Adaptive Process Planning and Scheduling in a Dynamic Sustainable Distributed Manufacturing Environment

Submitted in partial fulfilment of the requirements

for the award of the degree of

Doctor of Philosophy

by

RAMA KURTHI VEERA BABU

Roll No: 718029

Under the Supervision of

Dr. Vijaya Kumar Manupati
Assistant Professor, MED

Department of Mechanical Engineering
NATIONAL INSTITUTE OF TECHNOLOGY
WARANGAL – 506004
Telangana State, INDIA.

Aug - 2022

THESIS APPROVAL FOR Ph.D.

This thesis entitled **“Adaptive Process Planning and Scheduling in a Dynamic Sustainable Distributed Manufacturing Environment”** by **Mr. Rama Kurthi Veera Babu** is approved for the degree of Doctor of Philosophy.

____

____

____

Examiner

Dr. Vijaya Kumar Manupati
Assistant Professor, Department of Mechanical Engineering, NIT Warangal
Supervisor

Prof. A.Kumar
Head, Department of Mechanical Engineering, NIT Warangal
Chairman

NATIONAL INSTITUTE OF TECHNOLOGY

WARANGAL – 506 004, Telangana State, INDIA

CERTIFICATE

This is to certify the thesis entitled **"Adaptive Process Planning and Scheduling in a Dynamic Sustainable Distributed Manufacturing Environment"** submitted by **Mr. Rama Kurthi Veera Babu** for, Roll No. 718029, to **National Institute of Technology, Warangal** in partial fulfilment of the requirements for the award of the degree of **Doctor of Philosophy in Mechanical Engineering** is a record of bonafide research work carried out by him under our supervision and guidance. This work has not been submitted elsewhere for the award of any degree.

Place: Warangal.

Date:

Dr. Vijaya Kumar Manupati
Supervisor
Assistant Professor,
Department of Mechanical Engineering,
National Institute of Technology,
Warangal, Telangana State.

NATIONAL INSTITUTE OF TECHNOLOGY

WARANGAL – 506 004, Telangana State, INDIA

DECLARATION

This is to certify that the work presented in the thesis entitled **“Adaptive Process Planning and Scheduling in a Dynamic Sustainable Distributed Manufacturing Environment”**, is a bonafide work done by me under the supervision of **Dr. Vijaya Kumar Manupati**, Assistant Professor, Department of Mechanical Engineering, NIT Warangal, India has not been submitted for the award of any degree to any other University or Institute.

I declare that this written submission represents my ideas in my own words and where ever others ideas or words are included have been adequately cited and referenced with the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will cause for disciplinary action by the institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Place: Warangal. **Rama Kurthi Veera Babu**

Date: **Roll No. 718029**

ABSTRACT

Distributed manufacturing systems have become consensus, particularly networked manufacturing systems (NMS) owing to its flexible and adaptable nature in response to customized requirements. Highlighting of functions in NMS and their integration with recent key enabling technologies i.e., Artificial intelligence, Machine learning, Internet of Things, Block chain technology, and Agent-based techniques, etc. are of utmost importance to identify problems and increase its efficiency. Hence, besides the above-mentioned approaches, this work surveyed and analysed various articles systematically related to networked manufacturing in the context of knowledge creation and information, security, interoperability, and reliability. To identify the most related articles, the search has been conducted with Web of Science and Scopus databases. Subsequently, after evaluation 30 most related articles were selected and analyzed that further extended by identifying the issues and gaps in the existing empirical knowledge. Finally, thesis presents a roadmap for future research directions and developments.

Rising energy prices, increasing maintenance costs, strict environmental regimes have augmented the already existing pressure on the contemporary manufacturing environment. Although decentralization of supply chain has led to rapid advancements in manufacturing systems, finding an efficient supplier simultaneously from the pool of available ones as per customer requirement and enhancing the process planning and scheduling functions are the pre-dominant approaches still needed to be addressed. Therefore, this research aims to address this is-sue by considering a set of gear manufacturing industries located across India as a case study. An integrated classifier assisted evolutionary multi-objective evolutionary approach is proposed for solving the objectives of makespan, energy consumption and increase service utilization rate, interoperability, and reliability. To execute the approach initially, a text mining-based super-vised machine learning models namely Decision Tree, Naïve Bayes, Random Forest, and Sup-port Vector Machines (SVM) are adopted for classification of suppliers into task-specific suppliers. Following this, with the identified suppliers as input, the problem formulated as a multiobjective Mixed-Integer Linear Programming (MILP) model. We then proposed a hybrid multi-objective moth flame optimization algorithm (HMFO) to optimize process planning and scheduling functions. Numerical experiments have been carried out with the formulated problem for ten

different instances along with comparison of the results with a non-dominated sorting genetic algorithm (NSGA-II) to illustrate the feasibility of the approach.

Recent manufacturing systems did not just confine to optimal utilization of resources due to the global stance on strict environmental regimes. Collaborative effort to achieve sustainable practices in the decentralized manufacturing environment is a new complex problem. In this research work, with a networked manufacturing system we try to achieve both traditional as well as sustainable parameters by optimizing the performances such as makespan, machine utilization, and energy consumption. Thereafter, we formulate the problem as a mixed-integer non-linear programming (MINLP) model. To handle this NP-hard problem and to find the optimal solutions a Controlled elitist non-dominated sorting genetic algorithm (CE-NSGA-II) has been adopted. Finally, the results are analyzed with different scenarios to prove the proposed approach validation.

Secure, transparent, and sustainable distributed manufacturing system (DMS) is a pressing need for current Industry 4.0. In this research work, 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 Optimization Algorithm (HMFO) 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.

Keywords: Blockchain Technology; Network-based distributed manufacturing systems; Moth Flame Evolutionary Optimization algorithm; Smart Contract; Ethereum, data mining, Text mining, Sustainable distributed Manufacturing system.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

NMS	Networked Manufacturing System
IPPS	Integration of Process Planning and Scheduling
DMS	Distributed Manufacturing System
NSGA-II	Non Dominated Sorting Genetic Algorithm
CE-NSGA-II	Controlled Elitist-NSGA-II
CU	Customer User
EU	Enterprise User
MOP	Multi Objective Problem
MOEAS	Multi-Objective Evolutionary Algorithms
MU	Machine utilization
NG	Number of Generations
EC	Energy Consumption
MFO	Moth Flame Optimization
HMFO	Hybridized Moth Flame Optimization
HV	Hyper Volume
BCT	Block chain Technology
SC	Smart Contract
DME	Distributed Manufacturing Environment
MILP	Multi Integer Linear Programming
NP	Non -Polynomial

AKNOWLEDGEMENTS

I would like to express my sincere gratitude and profound indebtedness to **Dr. Vijaya Kumar Manupati,** Assistant Professor of Mechanical Engineering Department, National Institute of Technology, Warangal for giving me an opportunity to carry out doctoral work under his esteemed supervision. This work is a reflection of his thoughts, ideas, and concepts. **Dr. Vijaya Kumar Manupati** looks at things in the right perspective, and it has truly been a learning experience working with him. I owe a lot to him for making me a part of the continuity of the profession.

I extend my sincere gratitude to **Prof. N. V. Ramana Rao,** Director, National Institute of Technology Warangal, India for providing the necessary facilities and encouragement throughout my work.

I am thankful to **Prof. A. Kumar,** Head, Department of Mechanical Engineering, NIT Warangal and other faculty members for their encouragement and support extended during this period.

It's my great opportunity to express my deepest gratitude to the Departmental Scrutiny Committee members, **Dr. Selvaraj N,** Professor, Department of Mechanical Engineering, **Dr. Subhash Chandra Bose,** Associate Professor, Department of Mechanical Engineering, and **Dr. Suresh Babu E,** Assistant Professor, Department of Computer Science and Engineering, for their adeptness and many discussions during this research period.

I express my sincere thanks to my co-scholars **Samala Thirupathi, Malliswari Karanam,** NIT Warangal for their support and help in completion of this thesis.

I would like extend my heartfelt thanks to **Mr. M. V. Vijay Kumar** office staff of Mechanical Engineering Department, NIT Warangal for their constant help and encouragement.

A Special and deep gratitude to my father Ramakurthi Somaraju**,** my mother **Ramakurthi Satyavathi,** my brother **Ramakurthi Narasimha** for their unceasing sacrifices, endeavours and encouragement. I specially thank my wife **Varshini** for her emotional support during this journey.

Finally, I would like to acknowledge the help given by all the persons who have directly or indirectly supported the work.

Rama Kurthi Veera Babu

Chapter 1

Introduction

1.1. Introduction

The advancements in the information and communication technology brought enormous changes in the manufacturing sector and its applications. However, increase in concentration towards the product customization, advanced product functionality and, sustainability encourages the manufacturing organizations to operate in a distributed manner to gain the advantage in a highly competitive global market. Short cycle times, real-time information, constant flow of knowledge, short lead times, low logistics costs, etc. are the main requirements that modern manufacturing industries must meet in order to survive in global economic competition. Small and medium sized manufacturing units are familiar for their dynamic behavior, still they have several obstacles for their existence mainly due to their limitation in investing capacity, and to incorporate any new re-engineering activities required in their firm due to product complexity. Due to the reasons mentioned above, it is very difficult for Small- and Medium-size enterprises (SME) to withstand in a global competition. In addition, SMEs do not have sophisticated planning tools due to the above mentioned reasons [1]. This trends made the manufacturing units to explore for more opportunity by forming the SME as a network of enterprises. Recently emerged networked manufacturing or network based manufacturing system is one such distributed manufacturing paradigm that can support the above mentioned requirements and their functionalities.

1.2 Overview of networked manufacturing system

A Networked Manufacturing System (NMS) is defined as a system of manufacturing units that are geographically placed at a distances but connected with each other with the help of Internet and communication technologies to fulfill the needs of the customers. It has the capacity to express the essential features of manufacturing entities information and increase the scope for interoperability among the enterprises. The general structure of NMS is discussed in Fig. 1.1. A distributed manufacturing environment requires manufacturing companies to be flexible enough to meet changing customer needs, as well as provide the infrastructure for automation and communication to exchange data between different parts of the manufacturing system. These requirements are possible with the help of Networked Manufacturing System [1]. Initially, based on requirements and specifications, customer requests for products are processed through web mode, depending on the interaction of the customer with the NMS. From the Fig 1.1 explains the customer can make the request of the product in two ways. One path is through Customer User (CU) and other path is through Enterprise User (EU). Due to the nature of CUs, they accept product requests from customers and take the necessary steps to distribute data to potential and connected organizations according to product features and functionality via web services. However, on the other hand, after receiving a request from the client from the EU, unlike the CU, they have their own capabilities that can perform certain types of operations.

In order to carry out the work efficiently and with quality, activities that cannot be processed in the EU are selected from among potential organizations that can carry out these activities. From Fig. 1.1 while selecting the potential enterprises the EU or CU must have clear idea of the potential enterprises and their service capabilities. Here service indicates operation or task that required to produce a product. (Ex. Drilling, milling, grinding etc.). Here in our work we have considered the EU path to carry out the process. However, after collecting the product data and other information that require to manufacture the product, an effective approach was discussed with in the distributed environment. In addition, intense competition between enterprises and the availability of resources in different places to meet the production needs of each job entails the right choice of resources. Apart from achieving the higher performance and global optimal solution in a NMS, an effective scheduling and planning decisions should be made simultaneously [2]. In addition, an effective meta –heuristics has been used to solve the combinatorial problem of Integrated Process Planning and Scheduling (IPPS) to obtain the effective results for the objectives in the NMS. In very detailed manner to explain this, with advent of efficient technologies that can help NMS to fulfill the demands of customer particularly shifted from make to stock to make to order.

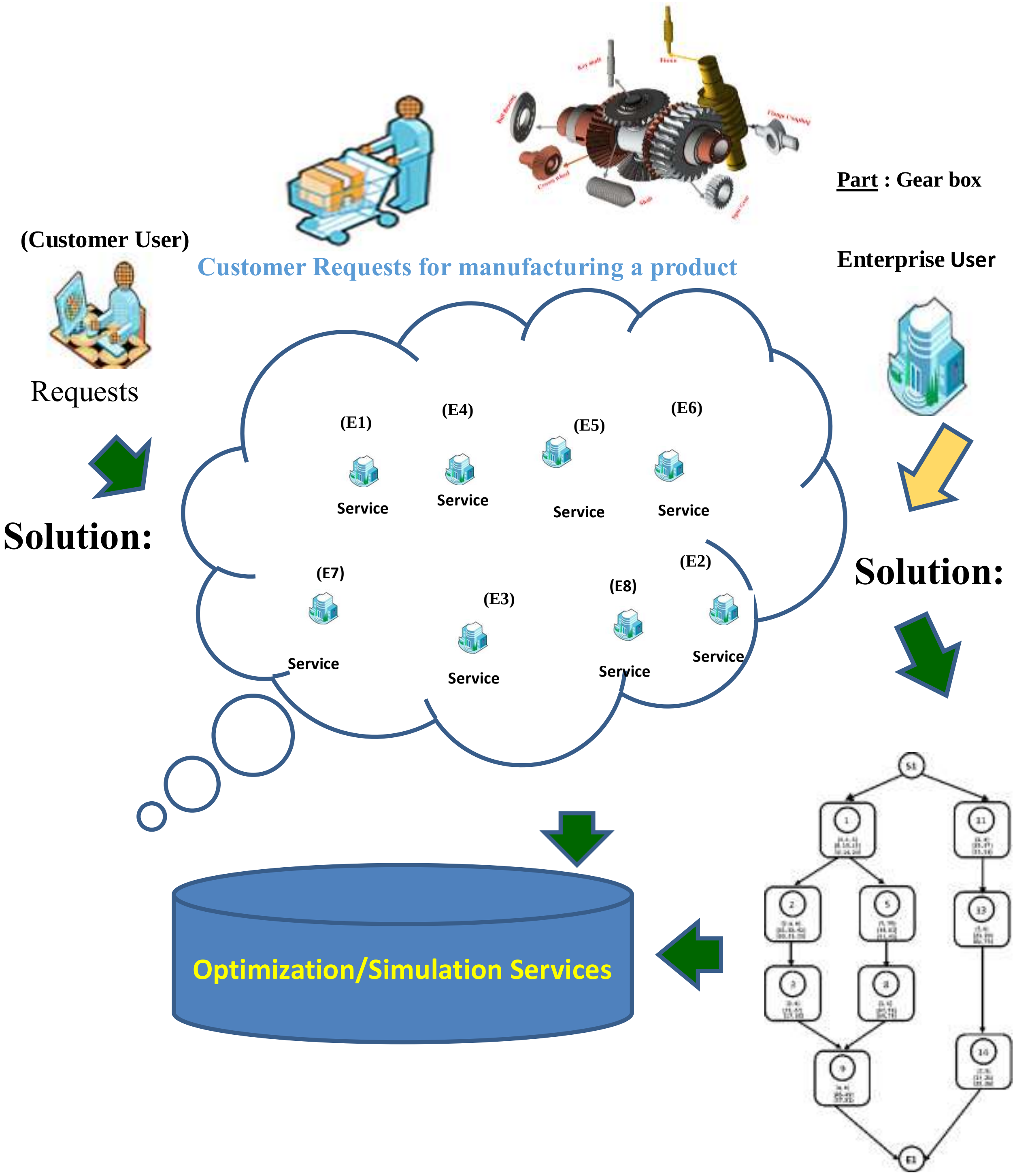


Figure 1.1 General structure for considered Networked Manufacturing System.

1.3 Definition of Networked Manufacturing system

Networked manufacturing enables integration of information and knowledge from product design to manufacturing and it shares resources between geographically distributed enterprises, thus endowing enterprises with the ability to respond to the market quickly [3]. In Networked Manufacturing System multiple job requests proposed by a customer with individual customization towards their product. Based on the collected requests job scheduling focuses towards the individual goals of each job. Moreover, in Distributed Manufacturing Environment (DME), the very nature of machines is that they are distributed at various locations to offer several services to complete a product. The NMS offer a plethora of opportunities to incorporate modern advance technologies to fulfill the modern day needs of individual customization of products.

1.3.1. Characteristics of Networked Manufacturing System.

The main characteristics of networked manufacturing are discussed here below.

- Networked manufacturing is obtained from the idea of collaboration of enterprises to form as a network to gain the competitive advantage in the market. It has the capability of providing information to the geographically distributed enterprises with the help of Information and Communication Technology (ICT) and other emerging technologies for Preprocessing of data, Process planning and scheduling, business process management with in the highly competitive global manufacturing environment.
- The NMS has the advantage of providing solution throughout the product life cycle.
- The Networked manufacturing environment provides added advantage of quick response to the customer request by increasing the competition among the enterprises.
- One of the major advantages with NMS, scope to obtain service reconfiguration feature with the rapidly changes in demand.
- The Networked manufacturing environment allows the manufacturing enterprise to remotely access to their services, that leads to reduction cost and time delays.

1.3.2. Advantages of network based Distributed Manufacturing Environment over conventional manufacturing environment.

The networked manufacturing environment differ from the conventional manufacturing environment in several ways that explained in the Table 1.1. The traditional manufacturing deals with Production scheduling and to improve the performance measures by managing manufacturing functions. In traditional manufacturing the scheduling mainly focuses to obtain overall optimal results for all the considered jobs not for the individual jobs. Whereas the NMS concentrates on obtaining optimal results or each individual jobs by considering various performance measures.

The main difference between the traditional manufacturing and Networked Manufacturing in the distribution of machines i.e., in traditional manufacturing machines are located in a single shop floor where as in NMS the machines are located at various geographical locations of several enterprises. Moreover, NMS each job can be processed on several process plans and it involves in multiple constrains that are changes quickly.

Table 1.1 Comparison of Networked Manufacturing System over Traditional Manufacturing System

Operations	**Traditional Manufacturing System**	**Networked Manufacturing System**
Range of operation	Restricted to a single shop floor	Distributed world wide
Functional duration	Ready to work over a constant time generally shift	Ready to work over throughout the clock
Firm structure	Function oriented	Project oriented
Organizational bonding between the firms	Competitive	Both competitive as well as Collaborating with each
Managing of manufacturing functions	linear	Integrated
Information and knowledge exchange	Happens in rare occasions	Highly co-operative to exchange of information and

1.4 Background Information of the considered Networked Manufacturing System research problem

The considered NMS have several customers and they order multiple products (jobs) based on their customized interests. These jobs can be denoted with ‘n’. (job 1, job 2, job 3, ..., job n). The scenario of considered NMS can be explained further with the help of the Fig.1.2. Each product can be produced with alternative process plans. Each process plans involves several operations. Similarly, to perform a single operation several alternative machines are available.

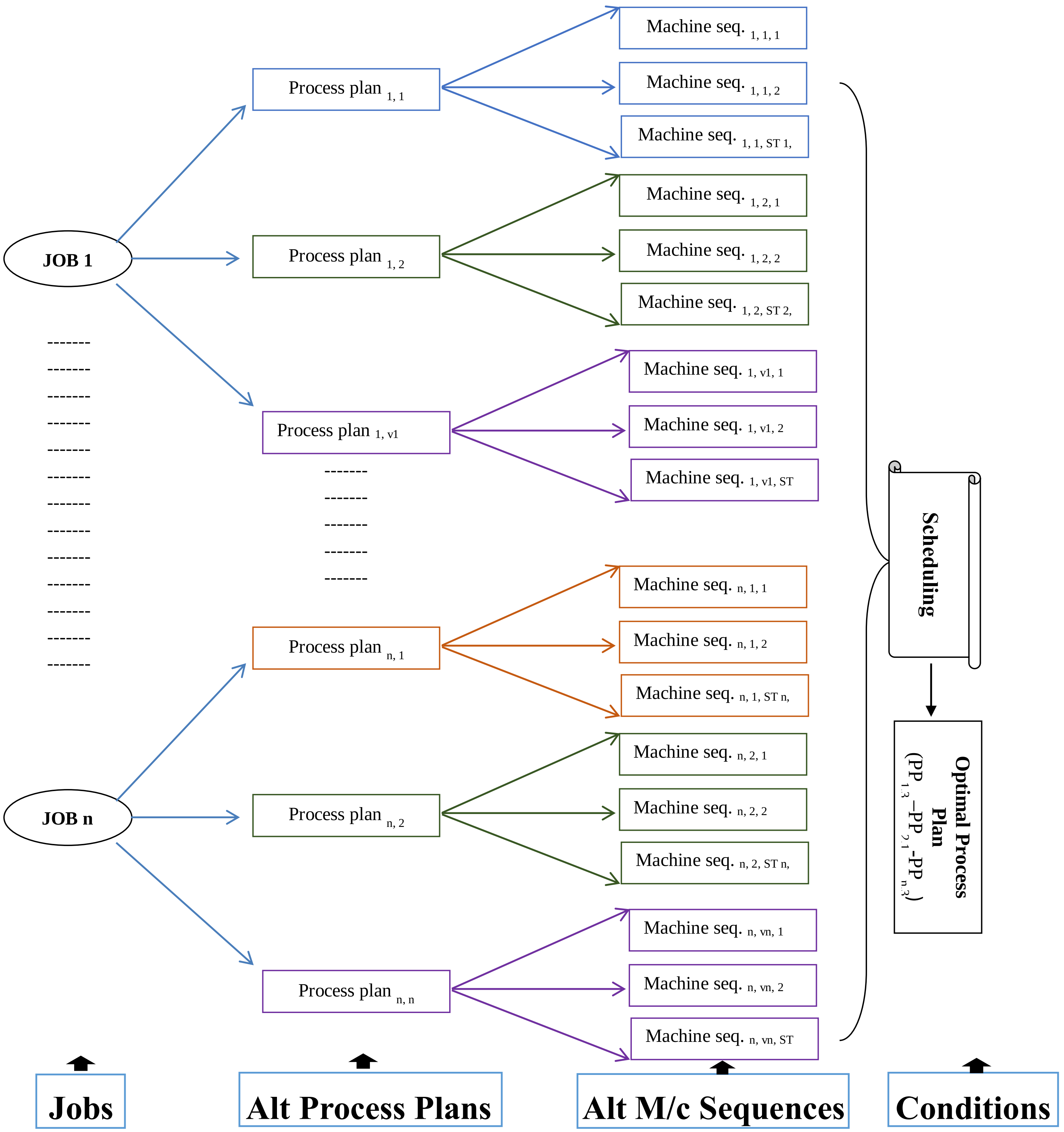


Figure 1.2 Proposed Framework for the considered Networked Manufacturing System research problem.

In the NMS these machines are located at various places to perform various operations on the products. Each machine is capable of performing operations required to make the products that are requested by the customer. In this regard several parameters important plays a role while assigning task to the machines effectively. In this thesis we considered various performance measures that highlight the sustainability of the NMS.

1.4.1 Definitions of several parameters used in the Considered Networked Manufacturing System research problem.

Process Planning: It consists of selection as well as sequencing of various processes and operations to convert a given raw material in to a finished useful product.

Scheduling: To assign the required operations to make a product in to corresponding capable machines without disturbing their sequencing in the given process plan can be called as scheduling.

Makespan (C max): The makespan is maximum completion time of all the jobs in a given system. It is the time required to complete the last job that leaves from the manufacturing system. This is the one of the predominant performance measures in any of the manufacturing system. Generally, lesser makespan is the better indicator to measure the performance of a manufacturing system.

Processing Time (P_{jk}): It is the time required for processing a job (j) on a particular machine k. Sometimes if the given job (j) is to be processed on any specific machines then the subscripts can be ignored.

Machine Utilization: It is the ratio between the effective processing time to machine running time. In other words, productive usage of the machine relative to its given capacity.

1.4.2 Various terms used Manufacturing scenario of process planning and scheduling problems

Single Machine: In the context of manufacturing system and shop floor environment single machine case is a special case and simplex of all forms of manufacturing scenarios.

Machines arranged in parallel: To perform a single operation given the flexibility to process on any two machines which are placed parallel and identical in their operation.

Machines arranged in parallel with parallel speed: To perform a single operation given the flexibility to process on any two machines which are placed parallel and not identical in terms of their working (or) operational speed to perform the given task.

Non-identical machines parallel: In this case the Non-identical machines are arranged in parallel which is an extend case of previous one mentioned above. The machines are completely different when they are placed parallel to each other.

Flow shop: Most commonly used term in the process planning and scheduling of manufacturing environment. With this kind of setup machines are arranged in series and every job has to undergo some operation on the machines that are arranged in series. However, the same path must be followed by all the jobs i.e. Job is undergoing operations on machine 1 first and then machine 2, and 3 and so on. There is a common assumption unless otherwise specified that all the jobs are following the queue in First come first serve order.

Flexible flow shop: It is a variant of flow shop, in this kind of setup, instead of machines are arranged in series, they are arranged in parallel. Here, the jobs followed same path i.e. Job is undergoing operations on stage 1 first then stage 2, and so on. At each stage job have a machine flexibility i.e. perform the one single operation more than on machine is available.

Job shop: Here the individual job has its own predetermining route to perform its operations to complete the job. In flow shop each job can only contact with on machine throughout its completion. Where as in job shop the job can contact with the machine more than once depending upon the requirement.

Flexible job shop: It is a minor modification of job shop instead of machines are in series, here in the flexible jobs shops, the work centers in series and in each stage of work center contains machines of identical in nature are arranged in parallel. Moreover, the same path must be followed by all the jobs i.e. Job is undergoing operations on work center 1 first then work center

2, and so on. At each work center job have a machine flexibility i.e. perform the one single operation more than on machine is available.

Open shop: In this kind of system each job has to be processed on each machine. However, some of the processing times can be zero. There are no limitations regarding to the sequence of each machine of each job.

1.5 Research issues

Several research issues were identified after a very detailed review on NMS as discussed below.

- It is required to integrate the process planning and scheduling to support networked based manufacturing environment.
- There is a need to design and develop different intelligent algorithms to solve the computationally complex multi/many-objective problem.
- There is a need to incorporate sustainability performance measures while process planning and scheduling in case of network based manufacturing.
- There is a need to improve the security aspects of networked based manufacturing environment.

1.6 Motivation for the Research

Motivated by the problem nature and its importance from the literature, the need to solve the complex problem in the NMS is one of the challenging task. Network Manufacturing offers several characteristics, opportunities and risks discussed below. Networked manufacturing contains several important characteristics like higher competition, short product life cycles, greater product diversity, fragmented markets, variety and complexity, customer oriented manufacturing. Apart from these mentioned characteristics, NMS offers several opportunities like flexible, agile, networked, adaptable, integral, interoperable, responsive to change, secure and transparent. Out of the above flexible, networked, secure are the very important criteria among all. In our current work more emphasis was given on the sharing of information in a secure and effective way between the networked manufacturing enterprises and their planning and scheduling function were optimized. Moreover, designing of suitable intelligent algorithms to solve the computationally complex (Non-Polynomial) NP hard problem while Integrated Process Planning and Scheduling (IPPS) to achieve the sustainability of the NMS.

1.7 Research objectives and scope of the thesis.

The present research pertains to the development and analysis of heuristics for the flexible process planning and scheduling problem. The overall framework of the present thesis, describing the objectives under consideration and the various methods used to achieve these objectives, is shown using Fig. 1.3. Based on the gaps identified in the existing literature, the objectives of the present research are summarized as follows:

- To develop text mining based approach for identification of appropriate enterprises in a networked manufacturing System.
- To formulate many-objective model along with constraints and analyze it with Integration of process planning and scheduling for a network based manufacturing system.
- To develop a sustainable networked manufacturing system using Non-dominated sorting genetic algorithm-II (NSGA II).
- To develop safe, secure and transparent transactions between enterprises using blockchain technology based approach for sustainable networked manufacturing system.

The scope of this dissertation is the application of various technologies that support Industry 4.0, such as machine learning (ML), artificial intelligence, blockchain technology, to improve safety aspects, as well as general productivity of a distributed production environment with optimal use of resources. Specifically, the machine learning assisted resource sharing and scheduling is covered in context of DME. The application of several evolutionary algorithms such as HMFO, NSGA II and, NSGA III to solve multi and many objective problems are concentrated in the current research. The major performance measures that were considered in the NMS to achieve the sustainability of the system. Moreover, the distributed technique i.e. Blockchain Technology applied to the considered research problem to improve the trust and transparency among the distributed enterprises.

In addition, the framework proposed in this study provides a clear understanding of data transfer and machine-to-machine communication. But methods to achieve it may have a wider field of research.

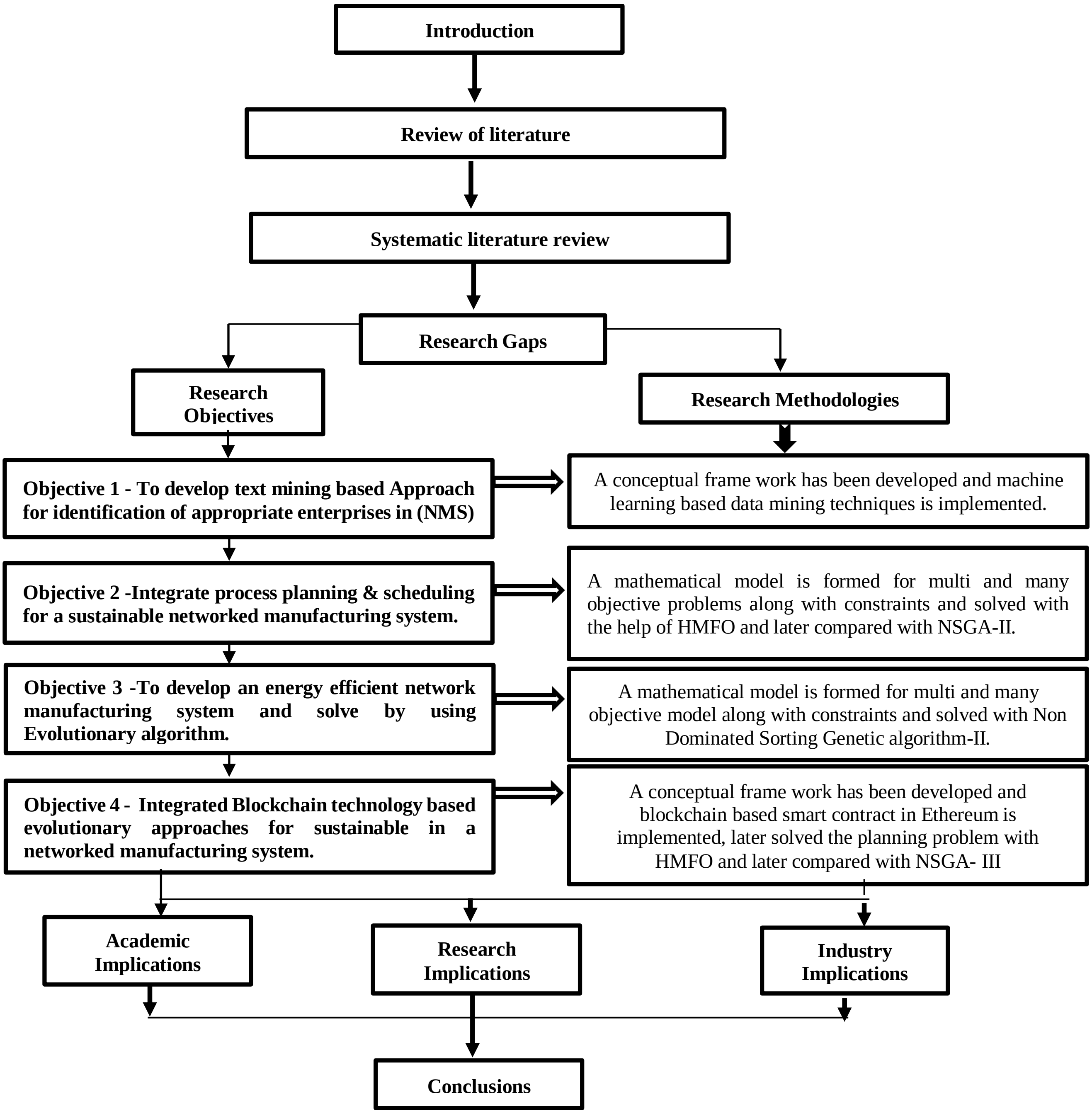


Figure 1.3 overall research framework of methodologies in the considered NMS.

1.9 Organization of the thesis

Chapter 1: In this chapter, the research preliminaries and the conceptual back ground of the research area has been explained. It also includes the motivation for the research work conducted and the scope of the research. The thesis is organized into seven chapters and the contents of each chapter are presented below in brief.

Chapter 2: In this chapter, the relevant literature on networked manufacturing with IPPS, Text Mining, and Blockchain Technologies are discussed in detail. Different categories of IPPS and their relevance with various manufacturing systems has been discussed. The need of the evolutionary algorithms for this research work is also discussed here and the classification of optimization problems for networked manufacturing is depicted with a framework. The gaps in the existing literature are identified in the present research.

Chapter 3: To identify the appropriate enterprises among the enterprises in a NMS is detailed with the developed text mining approach. In this chapter, to classify the suppliers a Text mining based approach has been implemented and a comparison study has been performed between various machine learning models such as Naive Bayes, Decision tree, Random forest, and Support Vector Machines. Later, with Task specific suppliers are used for performing the process planning and scheduling in context of NMS.

Chapter 4: Multi-objective model for IPPS for optimal process plan selection in networked based manufacturing. In this chapter, we have developed a mathematical model with minimization of makespan, maximization of machine utilization and minimization of energy consumption as objective functions in the presence of several constraints. It is a well-known NP-hard problem; hence an evolutionary algorithm is proposed to solve the problem with an illustrative example along with ten complex scenarios. The feasibility of the proposed approach and the algorithm is validated and the results are analyzed and compared.

Chapter 5: To develop a Multi-objective model in the context of sustainable NMS to integrate the process planning and scheduling for finding the near optimal process plans to select the multiple jobs. In this chapter, Mathematical model related to three objectives is formed with certain constraints and data is computed with practical variables. Lack of sustainability parameters in network manufacturing led us to solve the energy parameters of NMS and its new characteristics. Here, using Non-dominated sorting genetic algorithm (NSGA-II) algorithm the optimized process plan and scheduling for each job and Gant charts are drawn.

Chapter 6: Blockchain assisted multi-objective evolutionary algorithmic approach for resource sharing and scheduling in a sustainable distributed manufacturing system. In this chapter, an integrated block chain assisted process planning framework for distributed manufacturing systems has been proposed. Employing a Block chain-based smart contract to identify the potential enterprises and sharing of resources securely and transparently across the network and tested its feasibility with various considered cases. A multi-objective evolutionary algorithm-based Hybridized Moth Flame Optimization Algorithm (HMFO) is used to solve the considered problem in the scenario of distributed gear manufacturing industries. Later compared with many objective NSGA-III.

Chapter 7: Conclusions, contribution and scope for the future work

This chapter reports research contributions in the domain of dynamic process planning and scheduling. Managerial insights obtained from the analysis suggests that an adaptive flexible process planning and scheduling approach in a sustainable distributed manufacturing environment, and the developed methodologies discussed here can solve the requirements of distributed networked manufacturing environment in many ways.

Chapter 2

Literature review

2.1 Systematic Literature Review

The main aim of this chapter is to detail the solution approaches (or) methodologies applicable to distributed manufacturing systems. Particularly, issues related to interoperability, knowledge creation and sharing among distributed enterprises, and functional parameters integration i.e., Integration of Process Planning and Scheduling (IPPS) is detailed here. Subsequently, techniques such as datamining, text mining, machine learning, evolutionary algorithms (NSGA-II, Moth Flame Optimization (MFO), Hybrid-MFO, NSGA III) and blockchain implemented in this thesis work is discussed in detail. The above mentioned techniques, methods and approaches are used to conduct the Systematic Literature Review (SLR) to identify the gaps in the literature. Further, to define the objectives of this research work. The procedure of SLR is described in the subsequent sections.

The main aim of this section is to examine and synthesize a systematic literature review (SLR) to explain the roles of knowledge and Information management, Security, Interoperability, and Reliability in the Network Manufacturing System (NMS). In this present literature review, which is typically illustrative and incipient in nature, we predetermined to locate the big hurdles for knowledge management in the context of NMS. The review also aimed at examining how aspects like security and reliability are affecting maintaining networked manufacturing. Thus, the questions in this research for this SLR are RQ1: The role of knowledge and Information management, Security, Interoperability, and Reliability in NMS? RQ2: Which topics and functions are useful to Manufacturing functions management, distributed manufacturing systems, and response management are considered the scope of the NMS.

Additionally, this SLR also explains the importance of management of manufacturing functions like identification of resources, service management, etc., are discussed in the context of DMS. This SLR also covers the aspects like response management and service reconfiguration in the case of DMS's by: (i) reviewing in detail the effect of the mentioned referred areas in the NMS, with a special focus on knowledge and information management, interoperability, security, and reliability; (ii) Finding some of the research gaps in the reviewed literature; (iii) suggesting directions for further research.

The alignment of this present article is explained as follows: Section 2.2 explains the five-step based research methodology used in this article. Section 2.3 explains the locating studies of the SLR, and arrangement according to the conceptualization discussed in Framework. Section 2.4 reviews the obtained results, presenting valid research suggestions and guidelines for future work. Section 2.5 and 2.6 illustrates the limitations and conclusions of the present SLR.

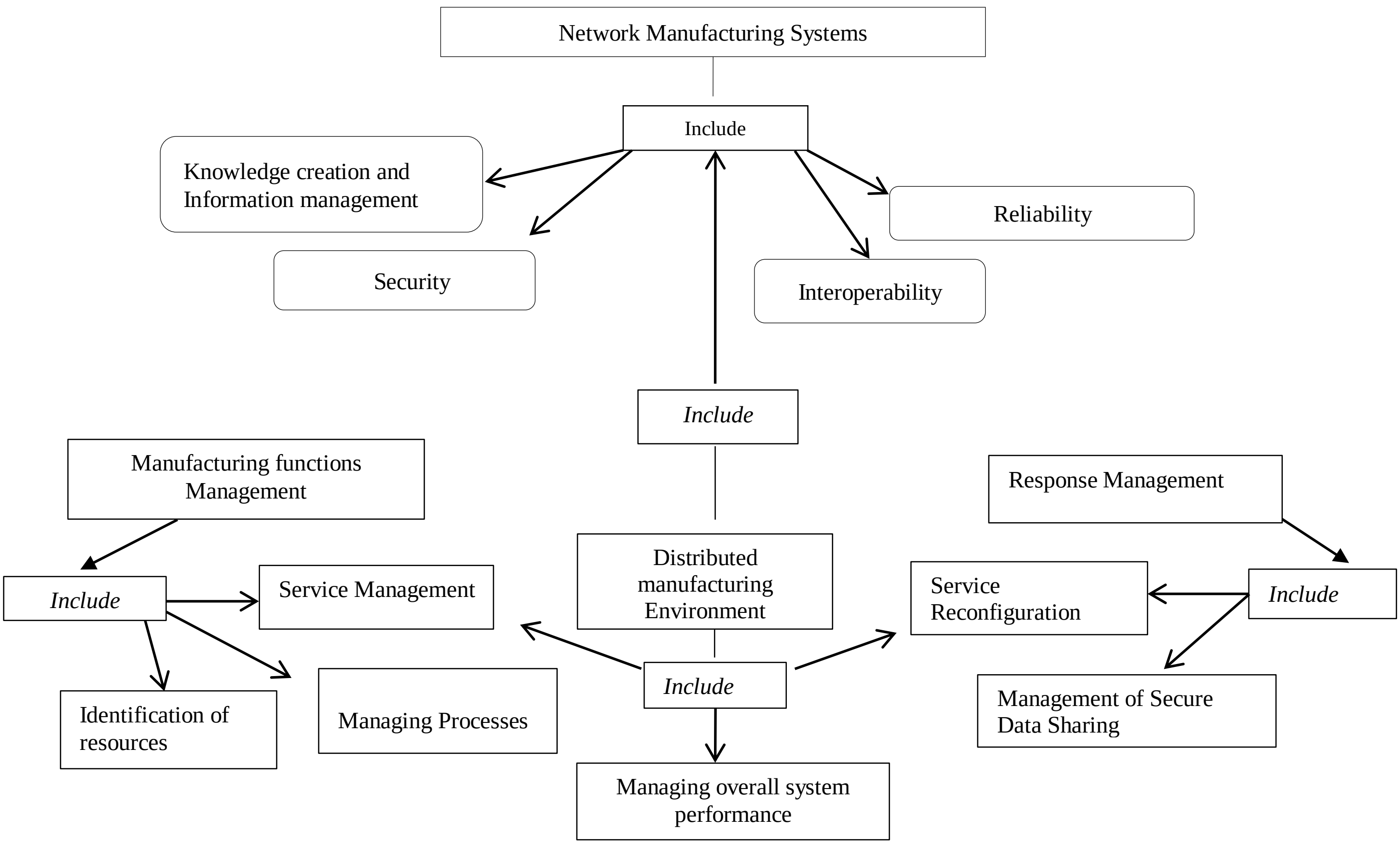


Figure. 2.1 Topics related to various aspects considered in the literature addressing the Network manufacturing systems.

2.2. Research Methodology of Systematic Literature Review (SLR)

This thesis adopted the systematic literature review (SLR) methodology [4] which is different from the traditional literature review. SLR methodologies allow constructing a framework for a thorough investigation of literature, adopting an organized and clear process [4]. As directed by [5], a scoping study of the area was performed before the SLR is to Plan the existing foundation for the work that needs to be done; (ii) enumerate the fit of the mentioned SLR into the present body of knowledge; (iii) describe concepts; (iv) decide the research queries to be answered. Therefore, this article systematically reviews related literature and their importance of knowledge and Information management, Security, Interoperability and Reliability in NMS. The adopted SLR consist of a five-step procedure, as suggested by [5] and [6]:

- Framing of the research question;
- Identifying the studies;
- Study selection and interpretation;
- Investigation and synthesis
- Reporting and presenting the results.

Step 1*: The mentioned research questions were formulated for this study*

RQ1**:** What is the role of knowledge and Information management, Security, Interoperability, and Reliability in Networked Manufacturing System?

RQ2: Which topics required for manufacturing functions management, distributed manufacturing systems, and response management are considered understudy of the Networked Manufacturing System?

Step 2: *Identifying the studies*

With the help of initial search strings mentioned in Table 2.1 the search was carried towards the research questions. This database is taken care of most of the research work that was published in Distributed Manufacturing, including almost all peer-reviewed articles on the area as shown in Tables 2.1 and 2.2.

Table 2.1 Search strings and displayed corresponding results from Web of Sciences.

Search String	Search field	Date of search	No of articles
"Distributed Manufacturing" (or) "Distributed-Manufacturing" and "Integrated Process Planning and Scheduling".	Topic	08-08-2019	183
"Distributed Manufacturing" (or) "Distributed-Manufacturing" (or) "Agent-based systems"	Topic	08-08-2019	761
"Distributed Manufacturing"(or) "Distributed-Manufacturing" and "Integrated Process Planning and scheduling" and "Agent-based method".	Topic	08-08-2019	30
Distributed Manufacturing" (or) "Distributed-Manufacturing" and Blockchain.	Topic	08-08-2019	21
Distributed Manufacturing" (or) "Distributed-Manufacturing" and Text- mining	Topic	08-08-2019	5

Table 2.2 Search strings and the displayed corresponding results from Scopus.

Search String	Search field	Date of search	No of articles
"Distributed Manufacturing" (or) "Distributed-Manufacturing" and "Integrated Process Planning and Scheduling".	Title-Abs-Key	08-08-2019	106
"Distributed Manufacturing" (or) "Distributed-Manufacturing" (or) "Agent-based systems".	Title-Abs-Key	08-08-2019	526
"Distributed Manufacturing" (or) "Distributed-Manufacturing" and "Integrated Process Planning and scheduling" and "Agent-based method".	Title-Abs-Key	08-08-2019	13
Distributed Manufacturing" (or) "Distributed-Manufacturing" and Blockchain.	Title-Abs-Key	08-08-2019	55
Distributed Manufacturing" (or) "Distributed-Manufacturing" and Text- mining	Title-Abs-Key	08-08-2019	13

Initially, a search string using the bibliographic database yielded in finding 183 articles for a search string Distributed manufacturing and IPPS. The number of results obtained for search string “Distributed manufacturing and Agent-based systems” is 761. Distributed manufacturing and Agent-based systems and IPPS search string resulted in 30 in number. Distributed manufacturing and “Text mining” number is 5. Finally, the search string “Distributed manufacturing” and “blockchain” resulted in 21 articles.

Step 3*: Study selection and interpretation*

In this section some filtering criteria are applied so that only the most related studies are added to the review. To pay much more attention to the latest studies, procedures and applications, initially, a 10-year time span is considered. Only articles that fall into the category of peer-reviewed publications in English were taken further [7] after filtering process. Further filtration has performed by restricting the search areas to Engineering, Manufacturing, Industrial, and Automation only.

The above exercise limited the papers up to 363. By using Mendeley and EndNote platforms, duplicates were removed by limiting the paper number to 280. This procedure has been carried out by several researchers to find the authenticity of work in those articles. Only some of the articles are eligible for review and must satisfy the three criteria.

I. Articles specifically related to the area of Distributed Manufacturing (it must include the studies that have done for IPPS)
II. Articles related to area of Text mining related to distributed manufacturing (excluding the areas like Health and Software fields)
III. Articles related to blockchain in case of distributed manufacturing (only related to the manufacturing applications). In this step, finally reduction of articles done up to 113.

Table 2. 3 Summary of the systematic review articles selection and evaluation.

Bibliographic database analysis	Search 1	Search 2	Search 3	Search 4	Search 5	Total
Web of sciences	185	761	30	21	5	1002
Scopus	106	526	13	55	13	713
Inclusive / Exclusive procedure for Web of sciences						
Time period	57	306	30	24	5	422
Document type	52	302	10	21	5	390
Research area	51	287	10	18	5	371
Language	50	284	09	16	4	363
Inclusive / Exclusive procedure for Scopus						
Date range	35	272	08	55	10	380
Document type	35	173	08	53	09	278
Research area	19	92	02	28	03	144
Language	18	89	02	21	03	133
After checking duplicates (in each search)	48	291	26	22	4	391
After checking duplicates (in all search)	32	263	07	41	3	366
Title and abstract evaluation	113					
Finally detailed article analysis	30					

Finally, 113 articles were filtered with related titles and abstracts. To perform the review in the most effective way a recent articles i.e. last 10 years were taken into consideration. In this process, some of the articles did not align with the subject area were filtered out and finally, 30 papers were listed out in Table 2.3. The comprehensive investigation of journals has been carried out in step 4, using two reviewers when their investigations and findings are useful to reduce the errors, to sort out any differences. Detailed information from these 30 articles was classified based on methodologies and important contributions. The later parts of the thesis describe the outcome of the current work in a systematic procedure.

2.3 Findings

Systematic literature review helps to deduce some findings from the existing literature. Based on the keywords the literature is discussed on each subheading in the next sections in 2.3.1 literature review on text mining, 2.3.2 Literature review on Integrated Process Planning and Scheduling in Distributed Manufacturing System. 2.3.3. Literature review on Block chain based manufacturing.

2.3.1 Literature review on Text mining

A review has been done for knowledge discovery and text mining techniques with data mining attributes namely depiction, and explanation, classification, estimation, grouping, and evolution in the domain of manufacturing [8]. An algorithm developed with K-means and support vector machine (SVM) clustering algorithms to examine the polarity of text and group the online hotspot detection forums into clusters depending upon their similarities [9]. An ontology implemented based on text mining strategy to extract the fault system information from the unstructured natural language text. All the information regarding to manufacture the product represented as a knowledge with the help of ontology named product ontology to share this knowledge in any platform by achieving interoperability [10]. At the same time Naive Bayes algorithm approach adopted for classification of manufacturing supplier. Later, Decision Trees and Random Forests utilized for supervised machine learning model based digit classification using Waikato Environment for Knowledge Analysis (WEKA) and have performed comparison on multiple performance parameters such as Kappa Statistic, Precision, Recall and F-measure [11]. A resource allocation strategy with help of big data and Machine Learning (ML) techniques is proposed to find the perfect forecasting of energy consumption patterns [12]. A text mining technique proposed for classification of sustainable environmental indices for service and manufacturing system. They also established relationships between indicator utility levels and company characteristics [13]. A support SVM classification algorithm presented to classify the supplier's text data of various web pages into manufacturing and non-manufacturing suppliers [14]. Adopted an e-commerce strategy for monitoring specific features of the enterprises. It describes how the records are obtained automatically from a corporate website using supervised classification algorithms [15].

2.3.2 Literature review on Integrated Process Planning and Scheduling in Distributed Manufacturing System.

Process Planning and Scheduling (PPS) mentions the requirement of manufacturing resources, operations and routes that are possible to manufacture a product and allocate the operations of all the jobs on machines, without disturbing the actual precedence relationships in the process plans [16]. In traditional manufacturing system, process planning and scheduling were carried out in a step by step process. To overcome the adverse effects with conventional way of PPS. Researchers have identified the need to integrate both PPS and have found the benefits of it in case of networked manufacturing environment. Manufacturing decision making (MADEMA) approach proposed for the assignment of work centre resources with multiple decision-making criteria in order to have an effective utilization for IPPS problem [17]. Later, Modified the above proposed MADEMA model consists of five basic steps and mainly focuses to find alternative machines and solve Integration of Process Planning and Scheduling (IPPS) problem [18]. Furthermore, a net-man strategic framework proposed where an operational mechanism is introduced for manufacturing organizations that helps to change their operations on a timely basis with the help of forming distributed manufacturing networks that helps as a performance enabler for manufacturers in Networked Manufacturing System (NMS) [19]. Agent based system has been proposed with distributed ruler method for distributed manufacturing systems as a function-based decomposition method to accomplish the Process planning and scheduling, and feasibility is illustrated through different case studies [20]. A Disruptive Innovation-Like Algorithm (DILA) is presented to minimize the tardiness for the job and to obtain optimal schedule for one machine varying setup times at irregular interval of time [21]. A novel way for formalizing dataset and concepts utilized to ontology. These were embedded in the product itself and hence made it interoperable in NMS. Furthermore, a two-level nested solution algorithm implemented by developing a hybrid adaptive genetic algorithm (HAGA) to achieve optimise process plans for multiple jobs in NMS. The feasibility of the approach is investigated through numerical experiments [22]. Binary Spring Search Algorithm (BSSA) based on simulation of Hookes law is employed to solve various optimization problems. Moreover, the results obtained by BSSA are compared with other standard binary algorithms namely grasshopper mechanism, bat algorithm etc. [23]. Mobile-agent based system introduced for IPPS in NMS, to prove the consistency of the proposed model comparison has been made with the Controlled Elitist Non-dominated sorting GA [24].

The effectiveness of Integration of production planning and scheduling approach is compared and proved with conventional sequential scheduling approach. An evolutionary algorithm based GA approach adopted for scheduling of integrated manufacturing and distribution systems [25]. A two loop algorithm consists of longest processing time rule based tabu search is proposed to obtain the optimal schedule in IPPS by minimizing the total cost for manufacturing and maintenance of machines arranged in series in case of NMS [26]. A hybrid dynamic-DNA assisted evolutionary algorithm proposed to solve N-person non-co-operative game in context to produce various optimal schedule for several jobs in a NMS [27]. A chaotic particle swarm optimization (C-PSO) algorithm for IPPS problem in network manufacturing and compared it with other benchmark algorithms like Genetic Algorithm (GA), Simulated annealing (SA), and hybrid algorithm to prove its superiority [28]. A Hybrid particle swarm optimization (H-PSO) described for IPPS and delivery route planning. This was implemented utilizing multi-purpose machines to minimize the cost and earliness and tardiness of the jobs [29]. A logic-based Benders decomposition (LBBD) algorithm mainly separates the decision variables into two sub-categories i.e. master problem deals with the process plan and sub problem deals with the sequencing has been emphasized that can solve the IPPS problem for finding an exact solution [30]. A Multi Integer Linear Programming (MILP) formulation proposed to minimize the storage cost and workforce cost in the airline industry. This problem mainly deals with the maintenance and repair operations allocation and it is an NP-hard problem solved by evolutionary algorithms [31]. A combination of algorithms namely H-PSO and GA with special operators have been presented to deal with the existed uncertainty in IPPS problem. Later Standard problems are considered to validate the effectiveness of the presented hybrid approach. [32]. To overcome the trapping of solutions to local optimum while solving the multi-modal functions by using several heuristic algorithms, a comprehensive learning particle swarm optimizer (CLPSO) has been proposed that combines the advantage of local search (LS) strategy along with the excellent global search ability of particle swarm optimizer (PSO). In this work several multi-modal benchmark functions like CEC2013 are tested for determining the effectiveness of CLPSO-LS algorithm [33].

A dynamic scheduling method proposed for cellular manufacturing systems approach was based on artificial intelligence for network environment [34]. Furthermore, developed generic mediator architecture for effective coordination and task planning in a Distributed Manufacturing Environment (DME) [35]. Moreover, a review has been presented on the articles related to game theory and optimization methods for several applications of problems.

Furthermore, classification into various categories where game theory is useful for increasing the effectiveness of optimization, and optimization methods useful to solve the game theory problems and, also a combination of game theory and optimization can be useful for efficient solving of other classes of problems. The proposed classification based on four criteria's mainly based on nature of optimization (classic or modern), based on the number of objectives (single or multi) and, based on the type of game theory [36]. Subsequently reduction of maximum completion time, tardiness and, production cost along with the optimal schedule is generated with the help of GA integrated with Gantt chart (GC) methodology for DMS. A case study of manufacturing scenario with six jobs and twelve machines considered and solved with the proposed GA-GC method [37]. Likewise, resource, management and, part agents are considered in a multi-agent-based system to make the decisions in a timely manner with proper co-ordination to generate optimal process plans in a distributed scheduling environment [38].

To make the proper choice of action without deviating from optimal strategy in an uncertain environment that exists in distributional robust optimization problems where the decision-maker is not sure about the distribution of uncertainty that exists in the problem. In such a situation, this work gives the insight to explore the alternative ways that help to find the distribution of uncertainty by the decision-maker based on the observations found from the experiments. Algorithms are proposed to find the local optima and derived a common evolutionary stable strategy to explore their convergence rate by using a mean estimate [39]. A memetic algorithm discussed for the minimization of makespan for a distributed assembly permutation flow-shop scheduling problem to obtain accurate results [40]. A multi-objective based mixed-integer programming model implemented with the consideration of makespan and total traveling distance as objective functions and proposed a GA-based heuristic approach to obtain the optimum results in virtual manufacturing cells (VMC) [41]. To solve the scheduling and maintenance planning simultaneously in DMS an evolutionary based GA is proposed and validated the performance by comparing with the other algorithm. [42]. An IPPS problem is solved with the Simulated Annealing (SA) approach that contains the added flexibilities of process, operation, scheduling to optimize the utilization of machines and production cost in a DMS [43]. A GA introduced to identify a configuration near optimum in a manufacturing network. The performance of GA derived alternate designs is held in comparison with the output of an intelligent search algorithm. A new approach named hybrid of Estimation of distribution algorithm (EDA) employed to increase the profit in forwarding supply chain and to reduce the carbon footprints in closed-loop supply chain network system [44]. A framework model presented for the feasibility and merits and their applications of complex networks in advanced manufacturing systems [45]. A PSO with hill-climbing approach proposed for minimizing the functions Makespan and energy consumption in DMSs [46].

To avoid the difficulty of applying the heuristics for combinatorial nature of problems like crude oil operation scheduling problem, initially, the problem is converted to an assignment problem related to tanks and distillers and later a chromosome is implemented that helps to further application of meta-heuristics like NSGA-II for optimization of refinery schedule. A case study of a china refinery with three distillers and ten charging tankers with multiple objectives have been considered and tested successfully [47]. A modified PSO developed to generate an optimal process plan and verified its performance through 5 independent experiments and comparison with other meta-heuristic algorithms in the domain of flexible process planning research. Recent advances of multi-objective genetic algorithms (MoGA) described with differential evolution (HSS-MoEA-DE) to solve several multi-objective scheduling problems in manufacturing systems [49]. A hybrid harmony search and genetic algorithm (HSGA) proposed for an integrated job maintenance scheduling problem in NMS [50]. A hybrid algorithm is proposed to solve the lot sizing and IPPS problem in case of plastic moulding industry. The proposed approach compared with the SA for test its effectiveness. [51].

2.3.3. Literature review on Block chain based manufacturing.

In this section, an extensive literature review is presented to explore the applications of 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 [52, 53]. An order matching model has been developed by considering the possibilities of manufacturers and requirements of retailers to fulfill the demand in DMS [54]. This model mainly depends 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 for distributed environments [55]. In order to find the advancements of the BCT, an extensive overview of state-of-the-art blockchain is presented and discussed its important components such as blockchain assisted IoT, BCT-assisted security, BCT assisted management of data, and their applications, and its demerits potential trends and challenges [56]. Where, few studies developed frameworks, architectures, and methods by considering the key functionalities in BCT 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 [57]. 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 [58]. A BCT framework along with future prospective research topics to enable meaningful scholarly engagement has been proposed [59]. Furthermore, a data carrier framework was developed, which emphasizes reducing contract deployment prices and regulate 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 [60]. A business process management framework is developed to explain the need for incorporation of BCT in the workflow composition and management to verify the trustworthiness of the business partner [61]. Later, several studies were carried out 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 [62]. Some authors have proposed a BCT approach that explains the trustworthiness of businesses of a firm is processed as text with the help of smart contracts [63]. A mining procedure is proposed to analyze a variety of data associated with a traceability system in cognitive manufacturing [64]. The feasibility of the BCT is explained in the supply chain as a case of the Thai Automotive Industry and a questioner-based survey and regression analysis model was constructed and analyzed [65]. Going one step further a Feb Rec Prototype has been introduced, they explained the concept of linking computing nodes and, CNC machines to demonstrate the possibility of connecting them on a distributed interoperable network [66]. Furthermore, a Blockchain supply chain system is developed for tracking and tracking products including their evolution in the manufacturing process with help of smart contracts [67].

A tool developed by researchers 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 [68]. Thereafter, a web application was implemented 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 [69]. To overcome the threats and limitations with present approaches followed in inventory management a BCT-based smart contract implemented to trace and track the spare part details from manufacturer to the supplier and end-customer [70]. It is noteworthy that a concept is presented for the usage of BCT-based smart contracts for sharing of resources in distributed manufacturing networks, moreover, the benefits and risks of the proposed concept are explained [71].

Several authors used algorithms namely a Branch and price algorithm [72], SA [73], and Chaotic PSO algorithm [74] in DMS. HS/4/IJMS a hybrid Harmony Search and GA [75], ACO [76] has been used to optimize the process parameters in 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 [77]. In more detail, a multi-objective Greedy-based non-dominated sorting genetic algorithm III (GNSGA IIII) is presented to solve the scheduling problems with Interfering Jobs. Furthermore, several instances were tested for comparison of proposed GNSGA IIII and Benchmark algorithms under various performance indicators [78]. An Evolutionary Algorithm (EA) recommendations system architecture is developed for effective selection of EA for solving the scheduling problems in DMS. [79]. 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. [80]

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, a survey has been conducted concerning the manufacturing system where sustainability across product life cycle 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 [81]. A detailed discussion presented over the importance of sustainability parameters in DMS and their reasons and insights for suitability in DMS has been discussed [82].

However, many researchers 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 by [83]. Some researchers suggested, traceability and transparency in BCT offer key benefits across different areas in a sustainable supply chain [84]. Apart from 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 [85]. A summary of literature on blockchain based smart contracts and their applications are mentioned in the Table 2.4.

Table 2.4 A summary of literature on blockchain based smart contracts and their applications are mentioned.

S. No	Author details	Application area	BC platform	Extent of Application
1.	Bogner et al., (2016)	Secure rented services using a mobile apps	Ethereum	Proposed a framework.
2.	Shermin et al., (2017)	Governance	Ethereum	Conceptual Explanation
3.	Hou et al., (2017)	Power vehicles	NA	Introduced a method
4.	Risus et al., (2017), Li et al., (2018),	Secure data sharing using Blockchain	NA	Proposed a framework.
5.	Sharma et al., (2018)	Automotive Industry in a smart city	BCT based DLT	Proposed a distributed framework
6.	Mondragon et al., (2018)	Supply chain of composite materials	BCT based DLT	Proposed a framework
7.	Yeh et al., (2018)	Payments using mobile	Ethereum	Introduced a method
8.	Zafar et al., (2018)	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., (2020)	Secure and transparent data storage	NA	Proposed a framework.
10	Xu et al., (2019)	Manufacturing Supply chain management	Ethereum	A design scheme for sharing of information is proposed
11	Lohmer et al., (2019)	Smart contract for sharing of resources in a DMS	NA	Conceptual Explanation
12	Wang et al., (2019)	Finance	Hyperledger Fabric	Developed a framework

13.	Mohanta et al., (2019)	Internet of Things	Ethereum	Developed a framework
14.	Kumar et al., (2020)	Cloud Manufacturing	Ethereum based (DLT)	Developed a framework
15.	Lu et al., (2019)	IOT, Security, Data sharing in BCT	NA	Presented the State of the art literature
16.	Alkhader et al., (2020)	Additive Manufacturing	Ethereum based smart contract	Implemented or tracing of products in the supply chain.
17.	Panja et al., (2020)	Online voting	Ethereum	Implementation
18.	Hasan et al., (2020)	Tracing of parts in the manufacturing Supply chain	Ethereum based Smart Contract	Implementation
19.	Leng et al., (2020)	Sustainability in blockchain	NA	Presented the State of the art survey.
20.	Zheng et al., (2020)	Comparison of smart contract	Ethereum, hyperledger fabric, corda.	Presented the State of the art survey
21.	Wu et al., (2022)	Supply chain management in manufacturing	BCT based DLT	Framework of BCT for supply chain integration.
22.	Wang eta al., (2021)	Security related smart contract	Ethereum	A Systematic Literature Survey
23.	Kamble et al., (2021)	Indian Automobile Industry	NA	Framework of BCT for supply chain integration.

2.3.4 Literature review on IPPS on sustainability in Distributed Manufacturing.

As mentioned above clearly our considered problem deals with the process planning and scheduling and requires optimization algorithms namely a Branch and price algorithm [86], Simulated annealing [87], and Chaotic PSO algorithm [74] in DMS. HS/4/IJMS a hybrid Harmony Search and GA [88], ACO [76] 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 [89].

In more detail, a multi-objective Greedy-based NSGA III) is presented to solve the scheduling problems with Interfering Jobs. Furthermore, several instances were tested for comparison of proposed GNSGA III [90] and Benchmark algorithms under various performance indicators. An Evolutionary Algorithm (EA) recommendations system architecture is developed for effective selection of EA for solving the scheduling problems in DMS [91]. A multi-stage heuristic method implemented that helps to solve energy-efficient scheduling problem in DMS and the effectiveness of the proposed method is compared with the genetic algorithm [92]. A summary of literature for various methodologies in case of IPPS in context of distributed manufacturing shown in Table 2.5. 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.

Table 2.5 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., (2009)	Integrated Process planning and scheduling (IPPS) in DMS	Mean flow time, Makespan	Simulated annealing
2.	Petrović et al., (2016)	IPPS in DMS	Makespan, Machine Utilization, Mean flow time	Chaotic particle swarm optimization algorithm
3.	Liu et al., (2020)	IPPS in DMS	Energy Consumption, Total tardiness	Heuristic-based two-stage approach
4.	Liu et al., (2016)	IPPS in DMS	Makespan	ACO
5.	Manupati et al., (2016)	IPPS in DMS	Makespan, Machine Utilization	Mobile agent based method.
6.	Menezes et al., (2017)	IPPS in DMS	Operational cost	Branch-and-price algorithm
7.	Cheng et al., (2020)	Scheduling problems with Interfering Jobs	Makespan	GNSGA III
8.	Zuo et al., (2020)	Scheduling	Makespan	Evolutionary algorithm.
9.	Lohmer et al., (2021)	Process planning and scheduling	NA	Systematic Literature Survey

2.4 Classification of optimization models for distributed manufacturing system

The manufacturing industries are distributed and carries operations to produce a combine product, it is very important to do the process planning and scheduling and solving such problem requires several methods. Particularly the problem requires IPPS and involves multiple jobs, alternative process plans, each process plan involves large number of machines. In this scenario obtaining optimal sequence and near optimal solutions is a complex task.

The current literature suggests that the IPPS functions in distributed manufacturing are computationally complex and cannot be solved in finite polynomial time (NP- hard) in nature particularly with multiple objectives. Moreover, to minimize or maximize the fitness function that improves the performance of the manufacturing system, it is highly suggested to select a proper methodology. For high sized industrial problems getting an exact solution may not be possible but possibility of getting near optimal solutions is very high. In that particular situation where large sized data problems can be solved with the help of heuristics. The various methodologies available for IPPS in case of networked manufacturing are classified and shown in the Fig 2.2.

As discussed above in order to solve the combinatorial problems, the solution methodologies are mainly divided in to two types i.e. exact methods and approximate methods. Even though the solution obtained by the exact method are optimal in nature but the computational time required for them is increases exponentially with number of variables. Therefore, in the current scenario of DMS problem deals with large number of decision variables may not be suitable to solve by exact methods. On the other hand, the approximation methods might give near optimal solutions to the problems that are NP- hard in nature. These approximation methods also known as heuristics that specifically give near optimal or sub optimal; solutions that entirely depends on the selection and success of chosen heuristics. In this context the pure optimality of the solutions is sacrificed up to some extant upon each other and leads to get near optimal solutions and reasonable computational times.

The approximation algorithms are further classified as shown in the Fig.2.2 into constructive heuristics and improvement heuristics. A sequence of steps executed to get the complete solution instead of improving the solution can be observed in constructive heuristics. This process involves in deterministic way or randomized way. Improved heuristics require initial solution get by randomly or by using other constructive technique.

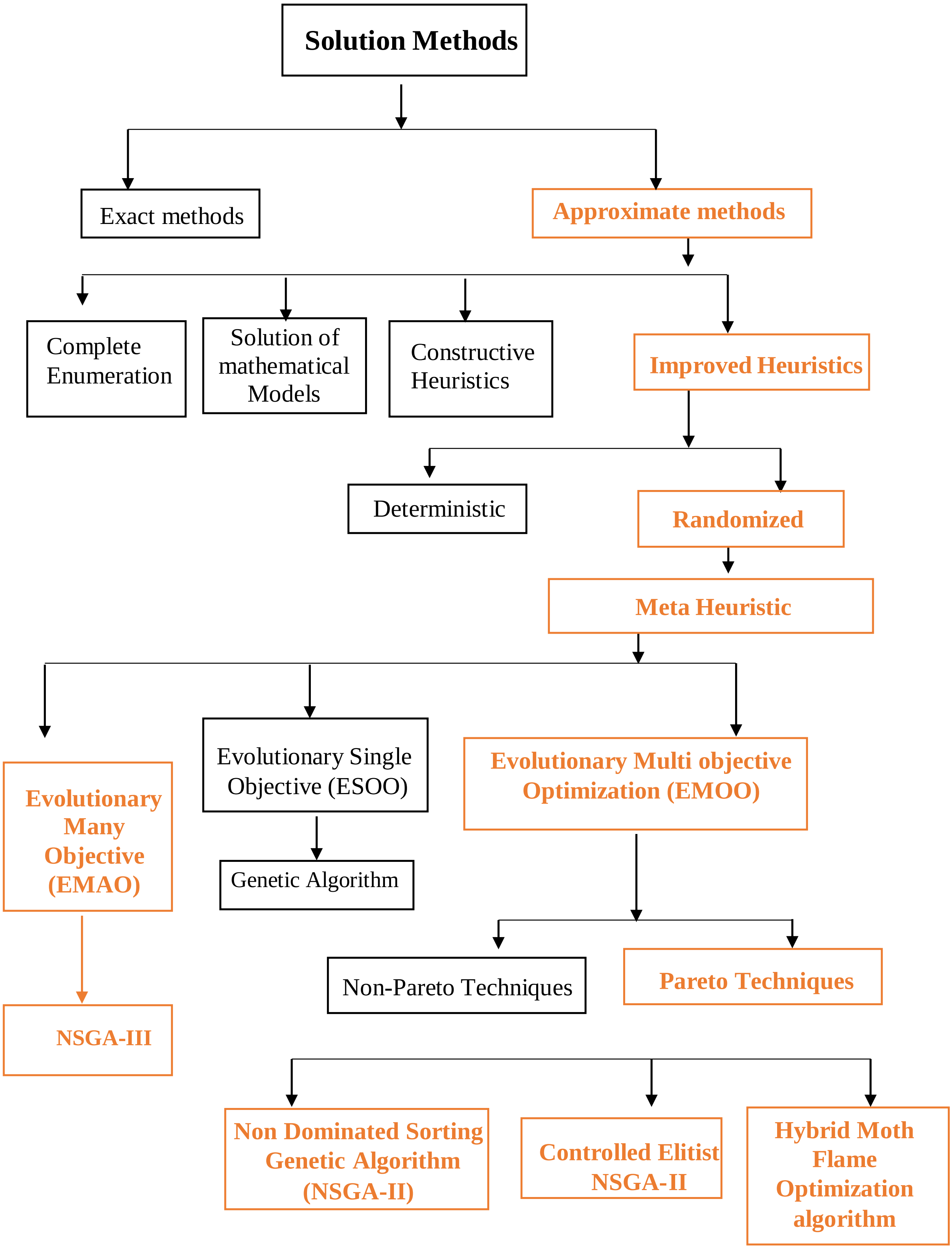


Figure. 2.2 Classifications of various methodologies used to solve IPPS in Networked Manufacturing System.

The considered networked manufacturing scenario where the problem is in NP hard nature to get the Integrated Production Schedules solved by a lesser computationally complex improved heuristic. Random search algorithms and randomized meta heuristics is one best path among the available paths to solve the problem. The meta heuristic further divided in to several types like single objective optimization (SOO), multi objective optimization (MOO), and many objective optimizations. Multi and many objective problems are generally deals with two or more objectives that can be conflicting in nature subjected to constraints. A colour indication mentioned in the Fig.2.2 details the methodology i.e. adopted in this thesis work. The detailed description of various methodologies for solving IPPS problems used in context of DMS are discussed in the sections. In addition to that we further discussed various advanced methods to share the information and to find the resources in DMS is discussed in the next sections.

2.5. Gaps Identified in the literature

The technological revolutions in the manufacturing industry made possible to react to the market changes particularly when the manufacturing based on the customer oriented manner. The exchange of information and resources between the distributed manufacturing industries must be transparent and secure. Moreover, finding the resources must be done in a simplified manner. In addition to that sustainability is one of the major concern of present day manufacturing systems. In the considered DMS obtaining optimal production schedules is of highly computationally complex in nature where large number of variables and constraints involved. To solve this NP-hard problems efficient evolutionary algorithms are required. The extensive literature helps to identify the gaps in the research related network based manufacturing systems.

- Advancements in technology, such as information and communication technologies (ICT) have changed the traditional manufacturing systems practices. This is especially true for a distributed manufacturing system due to its ability to cater to the needs such as Big data, interoperability, timely delivery, etc. which are small and medium scale in nature and are geographically distributed with the objectives as a selection of appropriate suppliers according to product type. Even though traditional methods or identification of suppliers is reported in the literature. But there is a need to adopt effective and efficient machine learning techniques for the supplier classification in DMS is identified as gap based on the literature.

- The flexibility and complexity of a distributed manufacturing environment create the need for investigating the multiple process plans and multiple performance measures. Hence, investigation of alternative process plans to the objective functions by acknowledging the NP-hard nature of the above scenario, there is a need propose a mathematical model and to solve the multi objective problem a suitable evolutionary algorithm is required. There is a need to design and develop different intelligent algorithms to solve the computationally complex multi/many-objective problem.

- Several performance measures were already considered in context of NMS in the past still. There is a need to incorporate sustainability performance measures while process planning and scheduling in case of network based manufacturing to minimize the negative impact of the industrial and ecological system.

- In the present network based distributed manufacturing system, the most of the information exchange takes place in the DMS in a traditional way, there is a high chance to get the cyber-attacks which results in a huge loss and even threat to the existence of networked based manufacturing systems. Apart from that there is need to maintain transparency among the distributed enterprises. In this regard there is a need for an effective technology which can offer transparency security while exchanging information between the enterprises.

In this chapter, a SLR has been explained that gives a clear cut observations of current status of the research and various methodologies adopted in the IPPS on networked based manufacturing. Through this extensive literature review research gaps were identified and based on those research objectives were defined for this thesis. In the coming chapters the defined objectives were detailed and their results are shown in chapter wise.

Table 2.6. Research gaps identified in the considered NMS.

Literature by authors	Research Gap
Gioele (2019), Alkahtani et al. (2019), Bianchi et al. (2018), Manupati et al. (2018), Park et al. (2017), Yazdizadeh (2015)	Text Mining approach for identifying relevant Enterprises is not considered in a network manufacturing.
Martin et al. (2019), Wang et al. (2019), Hockey (2019), Yang (2019), Claudio et al. (2019), Hung et al. (2019), Zhiyi et al. (2019), Martin et al(2017), Risjus et al. (2017), Sawichaya et al. (2017)	They have not considered the Security, Immutability, Transparency, Tractability aspects in case of networked manufacturing.
Barzanji et al. (2019), Xinyu et al. (2019), Kadir et al. (2019), Jigran et al. (2019), Mohammadi et al. (2019)	They have not consider the sustainability aspects while solving IPPS problem in case of networked manufacturing.

This SLR helps to identify various research gaps that are existed in the current research in the process planning and scheduling of the DMS. In this chapter various methodologies that were used to solve the scheduling problem are discussed in a detailed manner. The new methodologies that useful for solving the current problem are mentioned. This present research inclined towards the overall thesis of Adaptive Process Planning and Scheduling in a Dynamic Sustainable Distributed Manufacturing Environment. Here onwards in the next chapter work related to first objective has been presented.

Chapter 3

Text mining based supplier classification for effective resource and scheduling in distribute manufacturing.

3.1. Introduction

The Increasing competition, coupled with advancing computing technologies and the advent of decentralization in the supply chain has led to the attainment of a shorter product life cycle, reducing production costs and responding to customer demands with greater flexibility. Thus, manufacturing units are now leaning towards a distributed manufacturing environment far from the traditional approach for promptly manufacturing products [93]. This involves multiple processes consisting of classification of manufacturing units, assignment of tasks as per product category on the basis of requirements, information exchange within various units of an enterprise and between firms. All these together represent parameters of a compound scenario needed to be refined. In this chapter, the implications of the proposed classification and optimization/simulation-based integrated approach for the considered system is presented.

Managing supplier relationships, estimating the level of risk involved with various categories of suppliers, their capabilities, core services, constraints, target industries, and customers are some of the parameters for classification and selection. Dealing with selection of suppliers on such a large scale becomes tedious, thus it is required to subsume advanced techniques for supplier classification. This research incorporates text mining based on supervised machine learning models as one of the approaches for better resolution of the aforementioned supplier classification problem. Text mining is the operation through which information from unstructured text documents is extracted by devising non-trivial patterns and trends through statistical pattern learning. It contains various steps Viz. pre-processing, structuring of the input text data, extracting patterns and classification of information from various sources into a predefined genre [94]. Irrespective of any kind of manufacturing system, the prominent functions are process planning and scheduling. The former includes raw materials, semi-finished products, machine tools, and process information. The latter includes the allocation of resources over a set of constraints for manufacturing various entities [95]. Streamlining these functions through the hitherto approaches is not doable as these traditional approaches are characterized by linearity, the delay of time during the interval of planning and execution phase and single criterion optimization. In other words, it is estimated that 30 percent of the already generated process plans have to be modified in a given life cycle as a result of sequential processing of planning and scheduling in the existing systems [96].

This study seeks to address the following questions:

- How can supervised machine learning models be employed to find an efficient way for supplier classification, categorizing suppliers based on specific tasks?
- What are the benefits of the proposed approach on the considered problem and how these effects influence the manufacturing system in a real-time environment?

In this chapter, the context of the problem considered is a distributed manufacturing environment where different enterprises are geographically distributed, in which coordination and collaboration among such enterprises for mutual exchange of information without any significant loss of data represents a challenge. Motivated by the nature of the problem and the factors considered above, this study, in a distributed network manufacturing system pursues the classification, coordination, and communication to optimize scheduling and planning. This chapter introduces two different steps which have been duly scrutinized to achieve the desired optimization of parameters i.e., makespan, energy consumption, machine utilization rate and reliability of services. Initially, a supplier classification problem is explored under a manufacturing setup in which the suppliers are classified into task specific suppliers using text mining. The research contributions of this chapter can be outlined as given below:

- Proposing an integrated text-mining assisted process planning framework for distributed manufacturing systems;
- Comparison of various machine learning algorithms to test the feasibility of them.

In this chapter Section 3.2 discusses the problem description. Section 3.3 explains the proposed framework, algorithm for the text mining approach. In section 3.4 experimentation with a case study of the gear manufacturing industry is presented and the corresponding outcomes are explained in section 3.5. The chapter is concluded in section 3.6 by providing scope for future work.

3.2. Problem Description

Here, thirty-six medium-scale gear manufacturing industries located in a distributed manner across the southern part of India is considered as a case study for investigation for the prospect of providing an optimum solution with a composite multi-objective evolutionary algorithm approach assisted by a classifier. The systematic diagram of the gearbox that is being manufactured and their major parts i.e., gear, shaft, coupling flanges, key, bearing inner-outer race, bearing ball in a gearbox is mainly considered in this work shown in Fig. 3.1 have been considered for further investigation. Our research addresses areas of great concern related to finding a suitable supplier according to product consignment, interoperability, lack of efficient techniques, tools, and methods to enhance the productivity of the system. The process usually initiates with the customers in a network-based manufacturing service requesting resources through a particular supplier. However, the search and selection of an appropriate supplier is time-consuming with the consumer having little or no information about the respective capability narratives. We intend to categorize suppliers that manufacture gearbox related products in the market from their capability narratives and textual information collected via multiple product sourcing and supplier discovery platforms through text mining. Further, efficient supplier classification through the use of supervised machine learning algorithms are implemented.

The output obtained in the form of task-specific suppliers from the proposed supervised learning algorithm is fed as input for the considered network-based manufacturing system which consists of a set of job orders given by the customers denoted by n. Each job has numerous process plans by which it can be implemented. A set of available machines is distributed geographically to perform necessary operations in a process plan for the completion of the job. Considering the scenario of the current network-based manufacturing environment, the above background setting presents a challenge in terms of optimizing the objectives i.e., completion time, energy consumption, machine utilization rate, service utilization. As the problem is computationally complex and NP-hard in nature it becomes tedious to solve the above scenario. Thus, there is a necessity for an efficient and effective approach for an optimal process plan of the considered jobs. This research aims to provide a solution to approach an optimal process plan of the considered jobs by fulfilling the above-mentioned objectives. The scheduling problem discussed in the chapter 4. The output obtained by the machine learning method is used as an input for the scheduling problem.

Hence, an integrated machine learning-based evolutionary algorithmic approach where the outcome of a supervised algorithm is used as an input to an evolutionary algorithm is considered.

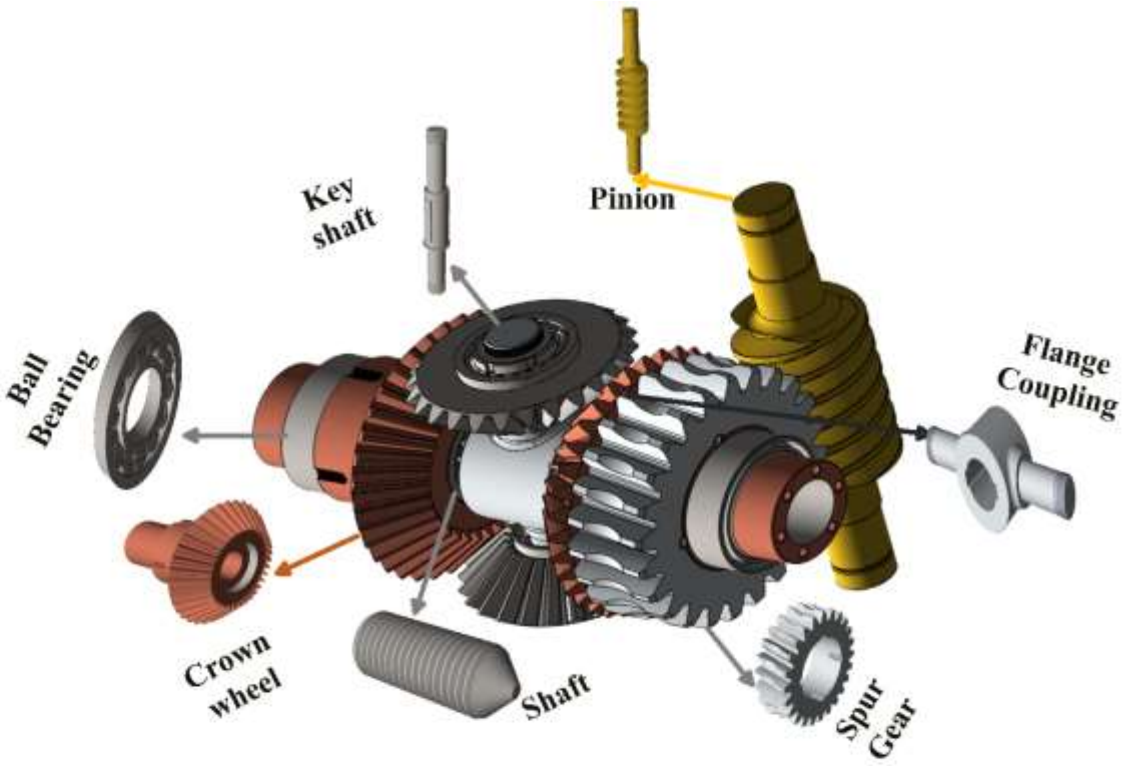


Figure 3.1 Various components in the gear box.

3.3. A Framework of the proposed classifier assisted evolutionary algorithm approach.

In this section, the proposed classifier assisted evolutionary algorithm approach as a framework is explained. A distributed manufacturing network environment is considered. Fig. 3.2 represents the proposed approach. In this model, the process gets initiated with the customer requests for a specific product. These requests are handled by enterprise user (EU) and customer user (CU) which are service providers in network-based manufacturing service. CU is an organization that accepts requests of different products from varied customers to complete the consignment agreement. To complete the accepted tasks, the available potential suppliers are evaluated from their database for assigning the task. Here, the main role of CU is to assign the tasks to the appropriate suppliers/manufacturers/distributors, etc., and to monitor their activities regularly to finish the task effectively and efficiently.

On the other side, EU also accepts multiple requests from customers; unlike CU, it has the capacity to provide some of the services by its own due to their own manufacturing unit. The remaining services will be fulfilled by assigning the task through potential enterprises as sub-contracting. In this study, we consider the EU service path where some of the services are fulfilled by itself. The next step would be the selection of the most appropriate supplier from the list of potential enterprises to whom the customer request must be forwarded. These suppliers are either maintained in the knowledge base or available as text corpus in the form of capability narratives. Initially, categorization into manufacturing and non-manufacturing units is carried out. Following this, text mining is implemented to finally perform supervised machine learning models based classification to differentiate the above dataset of suppliers into task-specific suppliers.

The above outcome of task-specific suppliers is further considered in a networked manufacturing environment to undergo three different stages i.e., order pool, task pool and service pool for processing requirements of the jobs to its final outcome. To execute the process of the desired product, requests are sent to order pool where the information of product specifications and their requirements are sorted and stored.

These stored orders in the order pool have classified the tasks into individual tasks by taking into consideration factors such as reliability, task priority, processing time, serviceability identified in this work. They contain a wide variety of tasks initiating from the procurement of raw goods to dispatching of the processed product to customers. Among the group of enterprises, the enterprises needed to meet the product and process requirements are chosen to carry out the manufacturing functions i.e., process planning and scheduling for optimal solutions.

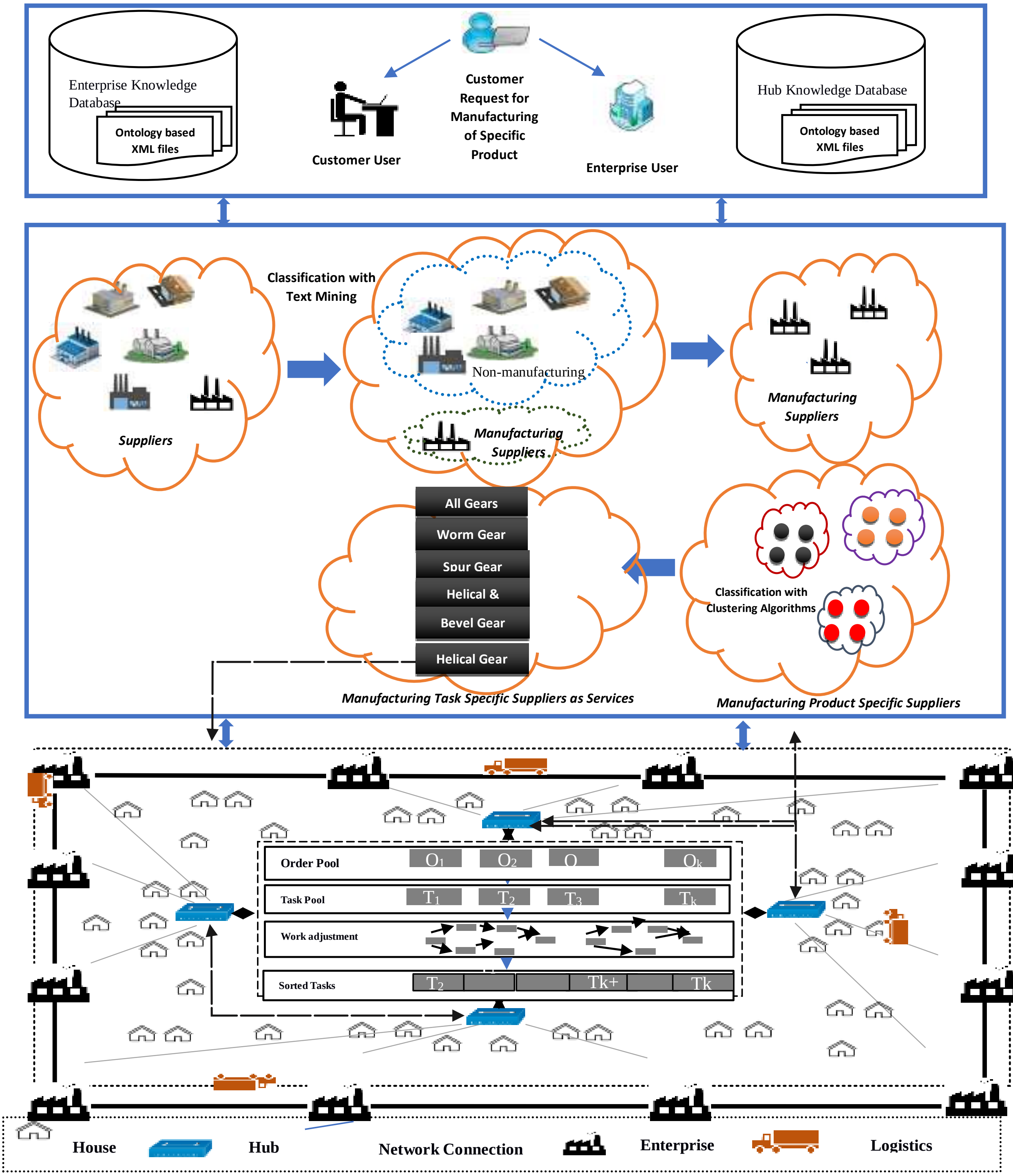


Figure 3.2 Framework of the proposed network manufacturing approach

Customers are having different demands related to their own specific product, after placing the order for the different product, the information related to the respective product is transferred into the order pool. After the orders are placed in the order pool, they are decomposed into separate tasks in the task pool.

In the order pool from various customers, the orders (O_1, O_2, O_3, $O_4 \ldots O_K$) are placed in the order pool to complete the order. All the orders are decomposed in individual tasks such as (T_1, T_2, T_3, …..T_K). For completing the one order various task are incorporated based on the services given by enterprises. From the very first stage of design, manufacturing, testing, packaging, processing, etc. is all the task are included in the respective product order. Various enterprises available to perform those tasks, task pool looks for the available services offered by enterprises. Based on the efficiency, reliability of the network, service time, service cost logistics cost, processing cost, etc. considering all the factors the enterprises choice is selected. The tasks are scheduled and sequenced amongst the enterprises according to the priorities (T_3, T_2, $T_1 \ldots T_K$, T_{K-1}) and requirement of the product. The procedure repeats for different orders depending upon the customer's requirements.

3.4. Experimentation part text-mining

3.4.1 Task-Specific Supplier Classification through Supervised Machine Learning Algorithms based on Text Mining

For the purpose of supplier classification, numerous suppliers representing the Gear Manufacturing Industry in India are taken as a case study. A flowchart explaining the methodology for supplier classification is shown in Fig. 3.3. After pre-processing and mining is applied to the above dataset, the suppliers are classified into manufacturing and non-manufacturing suppliers. Later, the manufacturing suppliers are further classified into task-specific suppliers with the help of various supervised machine learning algorithms. The performance of these algorithms is validated with different performance measures. The above approach is implemented using R and WEKA.

3.4.2. Creation of Supplier Corpus

Step1: For the purpose of text mining, a corpus of suppliers representing the Gear Manufacturing industry is created. This corpus is constructed with the help of capability narratives and textual portfolios accumulated via multiple product sourcing and supplier discovery platforms such as Thomas Net, Procure Search, Supply and Demand Chain Executive among several others.

Table 3.1. Various types of Gear Manufacturing

Category no.	**Type of gear**
Category 1	Bevel gear
Category 2	Helical and worm
Category 3	Helical gear
Category 4	Spur gear
Category 5	Worm gear
Category 6	All types of gear

The enterprises fall into 5 different categories of gear manufacturing as shown in Table 3.1. To accumulate any gear not falling into one of the above categories, a miscellaneous type 'All types of gear' is created. A test corpus of 40 different gear firms is also created to later validate our approach and classification performance. Out of these 4 are removed from corpus due to inadequate information on several parameters such as types of Gears, Types of Machines, Industries served, etc. This unstructured textual information is then read and converted into vectors in R.

3.4.3 Pre-processing of text corpus and Creation of Document Term Matrix

Step2: The prepared text corpus usually consists of delimiters, blank spaces, punctuation marks and stop words. These need to be removed before the application of machine learning models to remove any unnecessary bias during training. The corpus is thus subjected to data cleaning in this stage to eliminate the above entities. For proceeding to the later stages, two separate corpuses for manufacturing and non-manufacturing is created. The created manufacturing and non-manufacturing corpus are subjected to training and testing with varied weightages depending upon the frequency of their occurrence as explained [13].

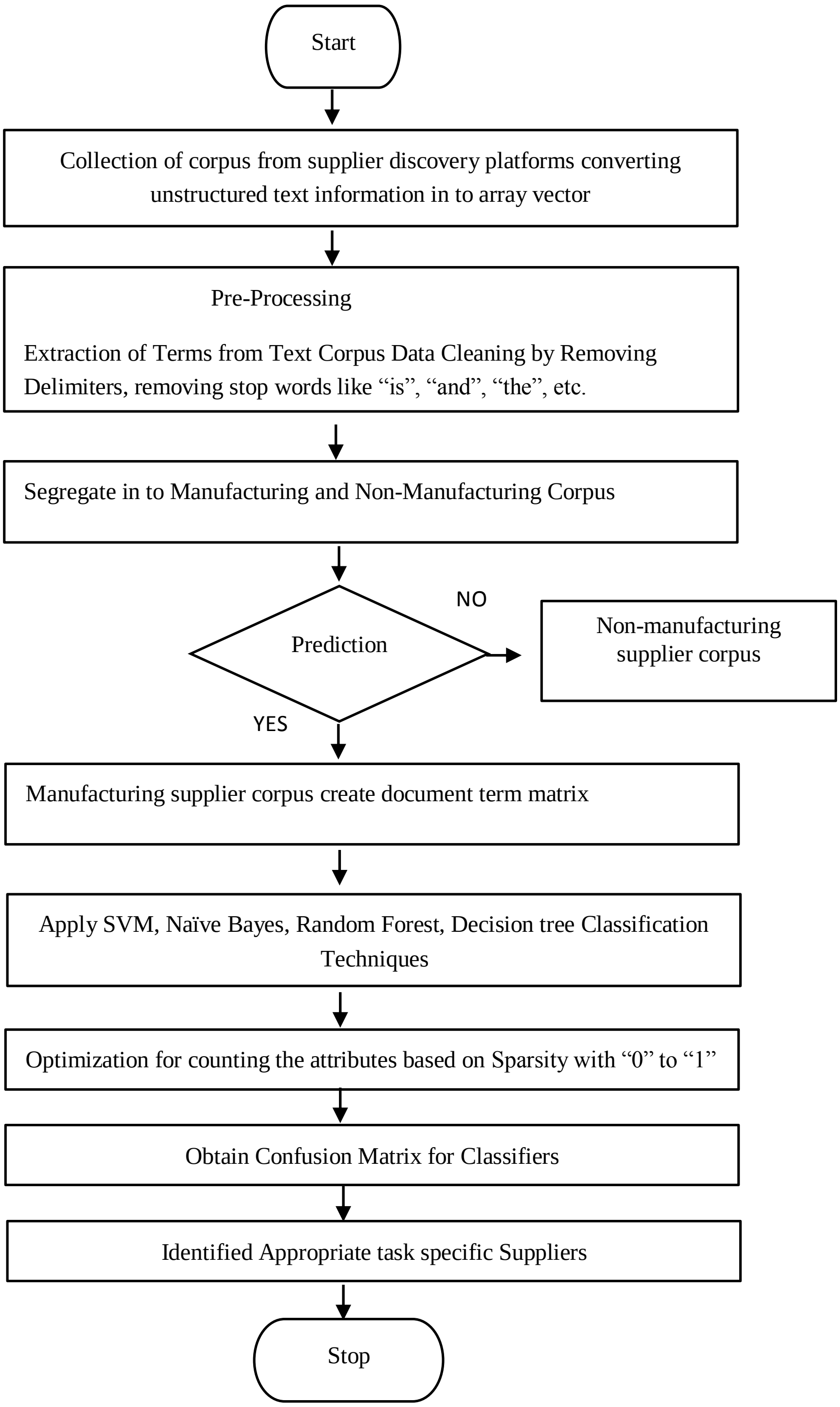


Figure 3.3 Flowchart for the proposed Text mining approach

The document term matrix is also created by selecting the features which are critical in the classification. The features can be represented in the form of a word cloud on the basis of varying sparsity measure which indicates the numeric occurrence estimate of the feature in the overall dataset.

This measure can then be used to eliminate those features which are not distributed entirely over the dataset and cannot be used to accurately classify the dataset. This matrix is then converted into a comma- separated value CSV file to be later be used in training and testing for machine learning models. Fig. 3.4 represents the word clouds which were formed for the six different gear classes with varying sparsity measure mainly Fig. 3.4 represents word Cloud for Worm at 0.77 sparsity, Fig. 3.5 Word Cloud for Spur 0.90 sparsity, Fig. 3.6 Word Cloud for Helical at 0.77 sparsity Fig. 3.7 Word Cloud for Worm and Helical at 0.90, Fig. 3.8 Word Cloud for Bevel at 0.90 and, Fig. 3.9 Word Cloud for All Types of gear at 0.77 sparsity are represented. Through hit and trial approach 0.77 was selected as optimum. Several industry-specific information is also extracted through the use of regular expressions as shown in Table 3.2.

Figure 3.4 Word Cloud for Worm at 0.77 sparsity

Regular expressions were utilized to segregate the raw data into the categories above. Also, Document term matrix has been created by selecting the features which are critical in the classification. The features can be represented in the form of a word cloud on the basis of varying sparsity measure which indicates the numeric occurrence estimate of the feature in overall dataset. This measure can then be used to eliminate those features which are not distributed entirely over the dataset and cannot be used to accurately classify the dataset.

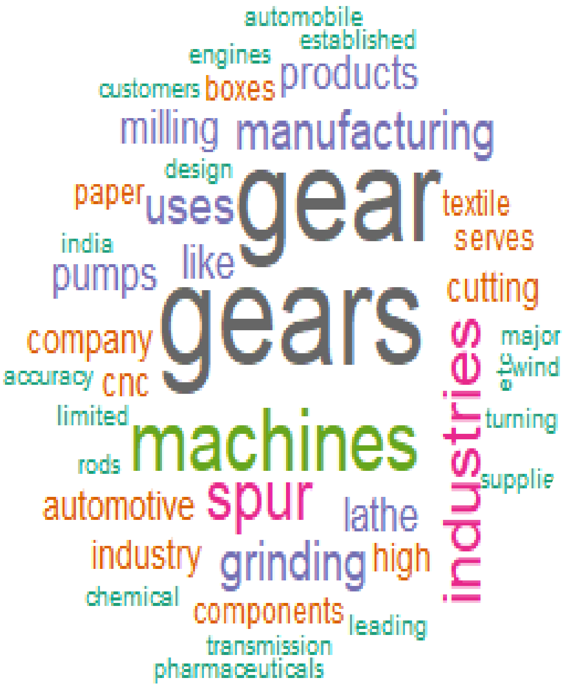


Figure 3.5 Word Cloud for Spur 0.90 sparsity

3.4.4 Classification into Task-specific suppliers

Step3: The manufacturing corpus represented through document term matrix in the format of comma-separated values is subjected to classification algorithms such as Support Vector Machines, Decision Tree, Naïve Bayes, and Random Forests to classify them into task-specific suppliers. This is implemented with WEKA [97]. Training is performed on a set of data set comprising numerous random capability narratives and testing on the above 36 capability narratives of gear manufacturing industries to classify them in one of the industrial categories. The performance of various classification algorithms is validated through the confusion matrix and other performance measures such as Kappa statistic, precision, recall, F-measure, etc., as shown in Table 3.3 obtained through WEKA. From Fig.3.10 shows Confusion matrix which gives an idea about the number of instances which are classified in various categories leading the classification to be either False positive (FP), True Positive (TP), False Negative (FN), True Negative (TN). Decision Tree is found out to be best among all models with 0.932 precision the least relative absolute error at 9.6%., followed by Naïve Bayes at 0.73. SVM and Random Forest have performed below par. Fig. 3.11 shows the results of the text mining depicting the enterprise information. It gives the detailed information of an enterprise such as types of gears manufactured, types of machines used, types of industries served and major clients of that particular enterprise.

3.5. Results and Discussions

Figure 3.6 Word Cloud for Helical for sparsity 0.9

The document term matrix is also created by selecting the features which are critical in the classification. The features can be represented in the form of a word cloud on the basis of varying sparsity measure which indicates the numeric occurrence estimate of the feature in the overall dataset.

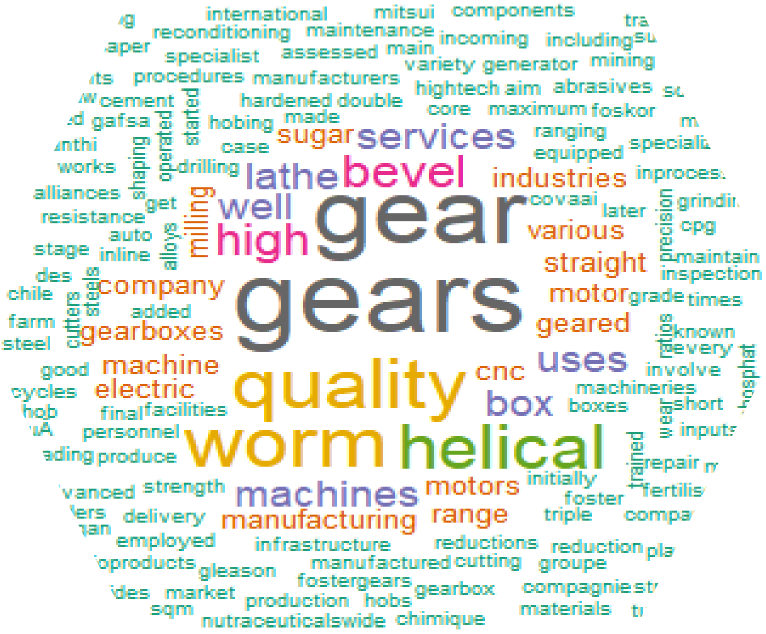


Figure 3.7 Word cloud for worm and helical at 0.9 sparsity

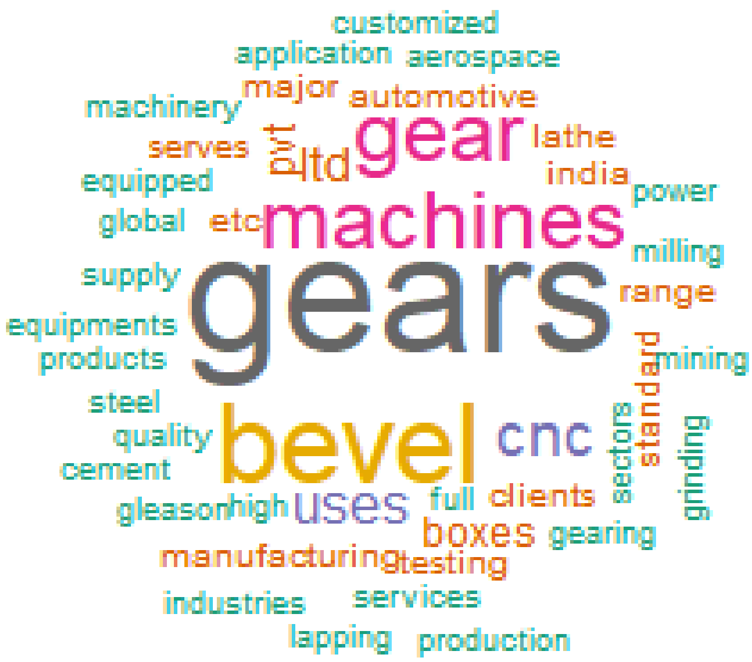


Figure 3.8 Word Cloud for Worm and Helical at 0.90 sparsity

Figure 3.9 Word Cloud for All Types of gear at 0.77 sparsity.

Table 3.2 Information to be extracted from mining and Gear Classification categories.

S.No	Information to be extracted from text mining
1	Types of machines (CNC, LATHE)
2	Types of operations (milling, drilling, grinding)
3	Types of gears (spur, helical, bevel)
4	Types of materials (steel, aluminium, bronze, brass)
5	Types of certifications (ISO 9000, ISO 14000
6	Types of manufacturing process (casting, forging, Extrusion)

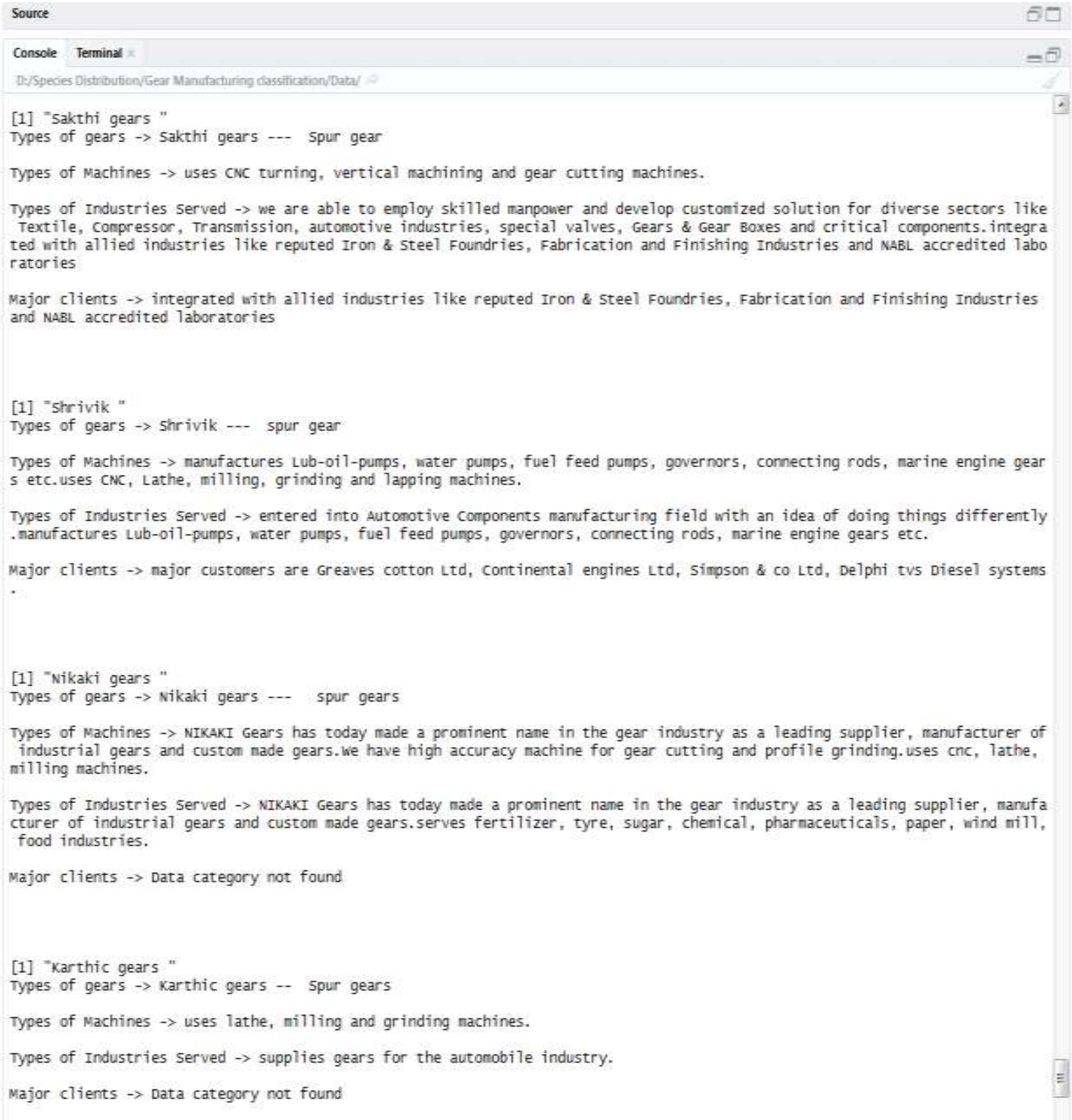
Naïve Bayes							Random forest							SVM							Decision tree						
a	B	c	d	e	f	a	A	b	c	D	e	f	a	a	b	c	d	e	f	a	a	b	c	d	E	f	←classified as
1	1	2	0	0	1	b	0	2	2	0	1	0	b	0	0	0	0	3	2	b	2	1	0	0	2	0	a—all types
0	5	1	0	0	0	c	0	6	0	0	0	0	c	0	4	0	0	0	2	c	0	6	0	0	0	0	b – bevel gear
1	0	2	0	0	1	d	0	0	2	1	0	1	d	0	1	0	0	1	2	d	0	0	4	0	0	0	c–helical &
0	0	0	6	0	0	e	0	0	0	6	0	0	e	0	0	0	3	3	0	e	0	0	0	6	0	0	worm
1	0	0	0	7	0	f	0	0	0	0	7	1	f	0	0	0	0	7	1	f	0	0	0	0	8	0	d – helical gear
1	0	1	0	0	5		0	0	0	0	0	7		0	0	0	0	1	6		0	0	0	0	0	7	e – spur gear

Figure 3.10 Confusion matrices of Naïve Bayes, Random Forest, SVM, Decision Trees.

Confusion matrix is the that is very much useful for comparison of various machine learning algorithms such as Naives Bayes, Random forest, support vector machine, and, decision tree in this work. In a confusion matrix row indicated the actual class and column indicates the predicted class. Confusion matrix is the important matrix in the field of statistical or machine learning field.

In the Fig. 3.10 out of all the above discussed algorithm decision tree is found to be best among others. For the decision tree from classes a to f most of the classes the predicted and actual classes are accurate which can be further seen by the Table 3.3. Furthermore, conclusions can be withdrawn with the help of a detailed values of F- Measure, True Positive (TP), True Negative (TN) values ROC area etc. are explained in the Table 3.3 very clearly.

Even though several other parameters are existed for comparing precession, recall, F-Measure are calculated based on the TP, TN, FP and, FN values to know the algorithm performance. The four algorithms namely Decision Tree, Naïve Bayes, Random Forest and, Support vector machine are applied on the present problem.



```
Source
Console   Terminal
D:/Species Distribution/Gear Manufacturing classification/Data/

[1] "Sakthi gears "
Types of gears -> Sakthi gears ---  Spur gear

Types of Machines -> uses CNC turning, vertical machining and gear cutting machines.

Types of Industries Served -> we are able to employ skilled manpower and develop customized solution for diverse sectors like
 Textile, Compressor, Transmission, automotive industries, special valves, Gears & Gear Boxes and critical components.integra
ted with allied industries like reputed Iron & Steel Foundries, Fabrication and Finishing Industries and NABL accredited labo
ratories

Major clients -> integrated with allied industries like reputed Iron & Steel Foundries, Fabrication and Finishing Industries
and NABL accredited laboratories



[1] "Shrivik "
Types of gears -> Shrivik ---  spur gear

Types of Machines -> manufactures Lub-oil-pumps, water pumps, fuel feed pumps, governors, connecting rods, marine engine gear
s etc.uses CNC, Lathe, milling, grinding and lapping machines.

Types of Industries Served -> entered into Automotive Components manufacturing field with an idea of doing things differently
.manufactures Lub-oil-pumps, water pumps, fuel feed pumps, governors, connecting rods, marine engine gears etc.

Major clients -> major customers are Greaves cotton Ltd, Continental engines Ltd, Simpson & co Ltd, Delphi tvs Diesel systems
.



[1] "Nikaki gears "
Types of gears -> Nikaki gears ---   spur gears

Types of Machines -> NIKAKI Gears has today made a prominent name in the gear industry as a leading supplier, manufacturer of
 industrial gears and custom made gears.We have high accuracy machine for gear cutting and profile grinding.uses cnc, lathe,
milling machines.

Types of Industries Served -> NIKAKI Gears has today made a prominent name in the gear industry as a leading supplier, manufa
cturer of industrial gears and custom made gears.serves fertilizer, tyre, sugar, chemical, pharmaceuticals, paper, wind mill,
 food industries.

Major clients -> Data category not found



[1] "Karthic gears "
Types of gears -> Karthic gears --  Spur gears

Types of Machines -> uses lathe, milling and grinding machines.

Types of Industries Served -> supplies gears for the automobile industry.

Major clients -> Data category not found
```

Figure 3.11 Screen shot of extracted enterprise information and classification into task specific supplier with text mining.

The Fig. 3.11 showing screen shot of extracted enterprise information and classification into task specific supplier based on the required features with was mentioned earlier with help of text mining.

Table. 3.3. Various performance measures for machine learning algorithms Decision Tree (J48), Naïve Bayes, Random Forest, and Support Vector Machines.

Decision Tree (J48)								
TP Rate	**FP Rate**	**Precession**	**Recall**	**F-Measure**	**MCC**	**ROC Area**	**PRC Area**	**Class**
0.400	0.000	1.000	0.400	0.571	0.604	0.881	0.694	All types
1.000	0.033	0.857	1.000	0.923	0.910	0.983	0.857	Bevel gear
1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	Helical and worm gear
1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	Helical gear
1.000	0.071	0.800	1.000	0.889	0.862	0.991	0.950	Spur gear
1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	Worm gear
0.917	0.021	0.932	0.917	0.903	0.899	0.979	0.923	Weighted Avg.
Naïve Bayes								
TP Rate	**FP Rate**	**Precession**	**Recall**	**F-Measure**	**MCC**	**ROC Area**	**PRC Area**	**Class**
0.200	0.097	0.250	0.200	0.222	0.114	0.490	0.181	All types
0.833	0.033	0.833	0.833	0.833	0.800	0.978	0.897	Bevel gear
0.500	0.125	0.333	0.500	0.400	0.316	0.672	0.573	Helical and worm gear
1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	Helical gear
0.875	0.000	1.000	0.875	0.933	0.919	0.938	0.920	Spur gear
0.714	0.069	0.714	0.714	0.714	0.645	0.808	0.723	Worm gear
0.722	0.046	0.738	0.722	0.727	0.681	0.838	0.750	Weighted Avg.
Random Forest								
TP Rate	**FP Rate**	**Precession**	**Recall**	**F-Measure**	**MCC**	**ROC Area**	**PRC Area**	**Class**
0.000	0.000	0.000	0.000	0.000	0.000	0.771	0.435	All types
1.000	0.067	0.750	1.000	0.857	0.837	1.000	1.000	Bevel gear
0.500	0.063	0.500	0.500	0.500	0.438	0.969	0.817	Helical and worm gear
1.000	0.033	0.857	1.000	0.923	0.910	1.000	1.000	Helical gear
0.875	0.036	0.875	0.875	0.875	0.839	0.996	0.986	Spur gear
1.000	0.069	0.778	1.000	0.875	0.851	1.000	1.000	Worm gear
0.778	0.045	0.669	0.778	0.717	0.692	0.964	0.898	Weighted Avg.
Support Vector Machines								
TP Rate	**FP Rate**	**Precession**	**Recall**	**F-Measure**	**MCC**	**ROC Area**	**PRC Area**	**Class**
0.000	0.000	0.000	0.000	0.000	0.000	0.500	0.139	All types
0.667	0.033	0.800	0.667	0.727	0.683	0.817	0.589	Bevel gear

0.000	0.000	0.000	0.000	0.000	0.000	0.500	0.111	Helical and worm gear
0.500	0.000	1.000	0.500	0.667	0.674	0.750	0.583	Helical gear
0.875	0.286	0.467	0.875	0.609	0.497	0.795	0.436	Spur gear
0.857	0.241	0.462	0.857	0.600	0.507	0.808	0.423	Worm gear
0.483	0.093333	0.454833	0.483	0.433	0.3935	0.695	0.380	Weighted Avg.

After the appropriate supplier/enterprise is shown to meet the product requests. The product requests are sent to order pool where the product specifications and requirements are stored. In task pool, the stored orders in the order pool are classified into individual tasks. They contain a wide variety of tasks starting from procurement of raw materials to dispatching of finished product to customers. The appropriate enterprises needed to meet the product demand is chosen in this stage. Taking various factors such as reliability, task priority, processing time, serviceability into consideration these individual tasks are sorted. These sorted tasks are sent to the enterprises for processing.

3.6 Conclusions

Advancements in technology, such as information and communication technologies (ICT) have changed the traditional manufacturing systems practices. This is especially true for a distributed manufacturing system due to its ability to cater to the needs such as Big data, interoperability, timely delivery, etc.

To execute the objective supplier discovery is implemented through text mining based on supervised machine learning models. The results of classification Decision Tree (J48), Naïve Bayes, Random Forest, and Support Vector Machines are validated through various performance measures mainly Precision, Recall, and F-Measures. Decision trees have been found to be best with a precision of 0.93 for the purpose. These selected potential suppliers and their related information have been transferred as input data to the next phase.

Chapter 4

Multi objective model for IPPS for near optimal process plans selection in a sustainable distributed manufacturing system

4.1 Introduction

Regardless of the type of production system, process planning and scheduling are important functions. The first includes raw materials, semi-finished products, equipment and technological information. The latter includes allocating resources across a set of constraints to produce various objects. These functions cannot be streamlined by the existing approaches as these conventional approaches are characterized by simplicity, planning and execution phase and time delay in the interval of single criterion optimization. In other words, it is assumed that 30 percent of already created process plans will be changed during this life cycle as a result of sequential processing of planning and scheduling in existing systems. In this work, the context of the problem considered is a distributed manufacturing environment where different enterprises are geographically distributed, in which coordination and collaboration among such enterprises for mutual exchange of information without any significant loss of data represents a challenge. Motivated by the nature of the problem and the factors considered above, this study, in a distributed network manufacturing system pursues the classification, coordination, and communication to optimize scheduling and planning. This work introduces three different steps which have been duly scrutinized to achieve the desired optimization of parameters i.e., makespan, energy consumption, machine utilization rate and reliability of services. Initially, a supplier classification problem is explored under a manufacturing setup in which the suppliers are classified into task specific suppliers using text mining. Thereafter, a multi objective mathematical model is developed to achieve the above-mentioned competing objectives. We then validate the viability of the above propounded approach by equating the outcomes with the proposed multi-objective Moth Flame Optimization (MFO). This work seeks to address the following questions:

- What type of mathematical model can be developed considering the optimization of various conflicting objectives such as completion time, energy consumption, interoperability, machine utilization rate?
- In what way can evolutionary algorithms be utilized to optimize scheduling and planning?
- What are the effects of the proposed approach and Evolutionary algorithm on the considered problem and how these effects influence the manufacturing system in a real-time environment?

4.2. Mathematical Modelling

The mathematical model involved is presented in the below section and its respective notations are explained in Table 4.1.

Table. 4.1 Notations used in mathematical model.

Notation	**Description**
E	The number of all the available jobs
G	The number of all the available machines
H_v	The number of all the available alternative process plans of job v.
Q_{vpk}	p^{th} alternative process plan for k^{th} operation of job v
S_{vp}	The Number of all the available operations in the p^{th} alternative process plan of the job v
L	Maximum completion time of v^{th} job from the all the available process plans
D_{vkpr}	For operation Q_{vkp} corresponding processing time of the on machine r
B	An arbitrary Integer which is a very large positive Integer.
C_v	The completion time till the processing of job v
C_{vkpr}	The earliest completion time till the operation Q_{vkp} on machine r
E_{vkr}	Indicates energy consumption for processing k^{th} operation of job v on machine r
Rel_{vk}	Indicates reliability of the k^{th} operation of job v

The present problem requires few assumptions that are very important to mention below.

1. Job Pre-emption is prohibited.
2. Until the previous job is completed, the successive job cannot be processed.
3. Only one job can be processed in an enterprise at a time.
4. The reliability of a machine with respect to time is constant and its value same for that particular machine for every operation while processing a job.
5. At time t=0, all machines and jobs are concurrently available.
6. The operations of every job and its respective sequence consisting of future processing tasks need to be pre-defined.

Decision Variables:

X_{vp} 1 The pth alternative process plan of job v is selected

0 Under other conditions

$Y_{vkpwtur}$ 1 The operation Q_{vkp} preceding over the operation Q_{wtu} on given machine r

0 Under other conditions

Z_{vkpr} 1 If given machine r is selected for Q_{vkp}

0 Under other conditions

Objectives:

$$\text{Minimization of makespan } (L_{min}) = \text{Max } C_{vkpr} \tag{4.1}$$

$$\text{Maximization of Machine Utilization } (U_v) = \frac{\sum_{v=1}^{E} D_{rv}}{\sum_{r=1}^{G} (mct_r - mst_r)} \tag{4.2}$$

$$\text{Minimization of energy consumption } (E) = \sum_{v=1}^{E} \sum_{k=1}^{S_{vp}} \sum_{r=1}^{G} E_{vkr} \tag{4.3}$$

Where D_{rv} represents processing time of job v on the r^{th} machine, and mct_r indicates finishing time of r^{th} machine i.e. the time taken to finish the final operation on r^{th} machine. mst_r is the start time of r^{th} machine.

Subject to Constraints:

The initial operation $(k=1)$ *in the possible process planpof jobv is mentioned as*

$$C_{vp1r} + B\left(1-X_{vp}\right) \geq D_{vp1r} \tag{4.4}$$

$$v \in [1,E],\ p \in [1,H_v], r \in [1,G]$$

The final operation for the possible process planpof jobvis mentioned below

$$C_{vpS_{vp}r} - B\left(1-X_{vp}\right) \leq C_{vpkr} \tag{4.5}$$

$$v \in [1,E],\ p \in [1,H_v], r \in [1,G]$$

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

$$C_{vpkr} - C_{vp(k-1)r_1} + B\left(1-X_{vp}\right) \geq D_{vpkr} \tag{4.6}$$

$$v \in [1,E],\ p \in [1,H_v], k \in [1,S_{vp}], r, r_1 \in [1,G]$$

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

$$C_{vpkr} - C_{wutr} + BY_{vpkwtur} \geq D_{vkpr} \tag{4.7}$$

$$v,w \in [1,E],\ p,u \in [1,H_v], k,t \in [1,S_{vp}], r \in [1,G]$$

Among the available process plans possibility to choose only one alternative process plan.

$$\sum_{v=1}^{E} X_{vp} = \begin{cases} 1 \text{ if processplan 'p' is selected from job'v'} \\ 0 \ \text{ otherwise zero} \end{cases} \tag{4.8}$$

$$p \in [1, H_v]$$

$$\sum_{r=1}^{G} Z_{vkpr} = \begin{cases} 1 \text{ if processplan 'p' is selected from job'v'} \\ 0 \ \text{ otherwise zero} \end{cases} \tag{4.9}$$

$$v \in [1,E],\ p \in [1,H_v], k \in [1,S_{vp}]$$

Table 4.1 presents the notation used in the mathematical model. Equation (4.1) to (4.3) represents an optimization of process parameters like minimizing makespan, maximization of machine utilization, minimization of energy consumption respectively. Precedence constraints of the operations are represented by Equations (4.4) and Equation (4.5), more specifically after the finishing of operation of a particular job only the next operation must start. Equation (4.6) represents different operations for the same job having precedence constraints that are unable to be processed simultaneously. Equation (4.7) represents that each machine can be able to process only one operation at a time and expressed as a constraint for the machine. Equation (4.8) indicates among the available process plans possibility to choose only one alternative process plan and Equation (4.9) represents that one machine only must be chosen for each operation.

4.3 Proposed Multi-Objective Evolutionary Algorithms

A nature-inspired population-based algorithm called moth flame evolutionary optimization (MFEO) Algorithm introduced by MirjalIli [98] that works on the consideration of the natural transverse movement of moths in nature. Moths can travel very long distances in a straight-line path. But interestingly along with the straight-line moths travel spirally near to the light sources which converge them in an optimized path to reach them to their destination. Based on this phenomenon Moth Flame optimizing algorithm is developed. The effectiveness of this algorithm over other algorithms (GA, PSO, ACO) is clearly shown by Mirjalili by considering several benchmark functions and case studies [98]. In this work, a hybridized form of moth flame evolutionary algorithm (HMFO) is presented. We have considered non-dominated sorting Pareto approach for the proposed algorithm hybridization. This has been implemented to the moth flame optimization with non- dominated sorting and crowding distance operators, and a flowchart for the same is presented in the next section.

Table 4.2 Initialization of parameters for proposed solution algorithm.

Process Parameters	HMFEO	NSGA II
Population Size/No of Moths	200	200
Number of generations	1500	1500
Mutation Probability	-	0.07
Cross Over Probability	-	0.76

The parameters for the HMFO technique are specified for the implementation of algorithms shown in Table 4.2 with the number of moths as 200 and the maximum number of iterations as 1500. Upper boundary and lower boundary values are specified based on the test data input.

Step 1: In HMFO 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.

$$L = \begin{bmatrix} L_{11} & L_{12} & \ldots. & L_{1d} \\ L_{21} & L_{22} & \ldots. & L_{2d} \\ L_{31} & L_{32} & \ldots. & L_{3d} \\ L_{41} & L_{42} & \ldots. & L_{4d} \end{bmatrix}$$

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. 4.1.

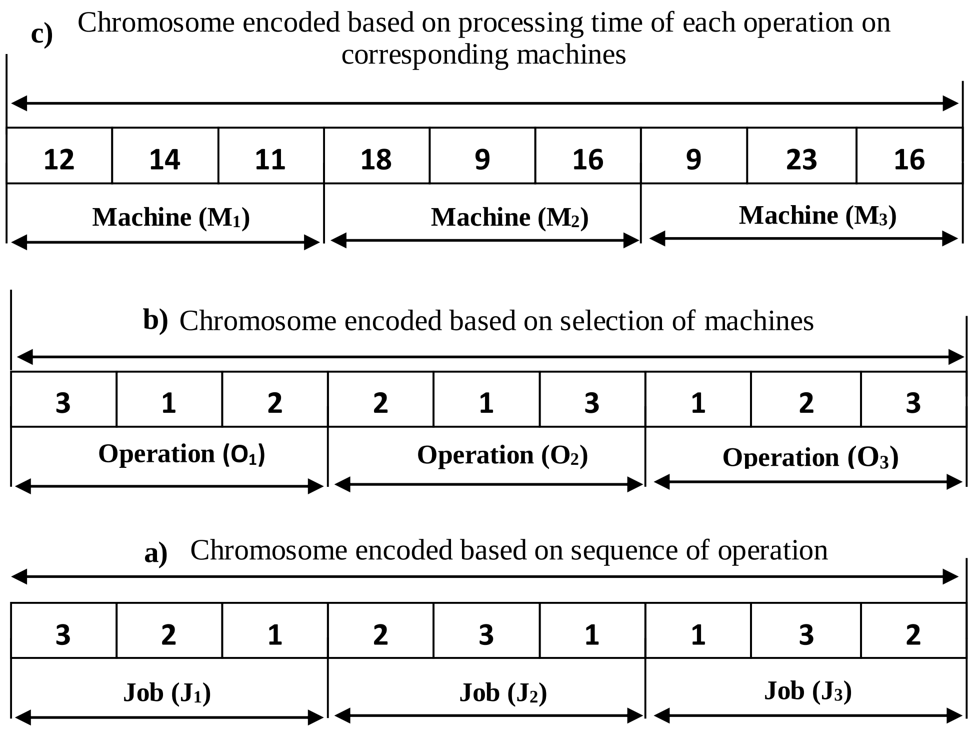


Figure 4.1 Representation of chromosome initialization for make span.

The example of encoding schema represented in Fig. 4.1, the encoding consists of three parts. If we observe from bottom to top, Fig. 4.1 a) is encoding based on a sequence of operations of each job., which can determine the sequence of operations needed to produce a job. Fig. 4.1 b) is encoding based on machines, which can choose the machine for each operation. Fig. 4.1 c) is encoding based on the processing times of each machine for the corresponding operation. Therefore, a chromosome in Fig. 4.1 shows 3 jobs, which consist of 9 operations, and will be processed on 3 different machines. The processing sequence of this chromosome can be represented as $Q_{ik}{}^{j}$ that the j^{th} operation of the i^{th} job will be processed on the k^{th} machine. Where $Q_{23}{}^{1}$ is the 1^{st} operation of the second job will be processed on the third machine. Based on the encoding schema time matrices were obtained and, the time matrices for makespan can be represented as follows. Makespan=zeros(mach, opns, pp, jobs);, makespan(:,:,2, 3) matrix below indicating the processing time values of machines for the corresponding operation for the 3rd job and 2ndprocess plan. The remaining values in the matrix are kept as zeros.

	M1	**M2**	**M3**	**M4**	**M5**	**M6**	**M7**	**M8**	**M9**	**M10**	**M11**	**M12**
O_1 [	12	18	9	0	0	0	0	0	0	0	0	0;
O_2	14	9	23	0	0	0	0	0	0	0	0	0;
O_3	11	16	16	0	0	0	0	0	0	0	0	0;
O_4	0	0	0	0	0	0	0	0	0	0	0	0;
O_5	0	0	0	0	0	0	0	0	0	0	0	0];

Step 2: based on the above, the energy consumption matrix (Equation (4.10)) is obtained through multiplication with the corresponding time matrix with specified energy consumption input mentioned in Table 4.3. The encoding schema is specified in Fig. 4.2.

EC matrix (o, m, p, j) = L matrix (o, m, p, j) * E (Rated energy matrix); (4.10)

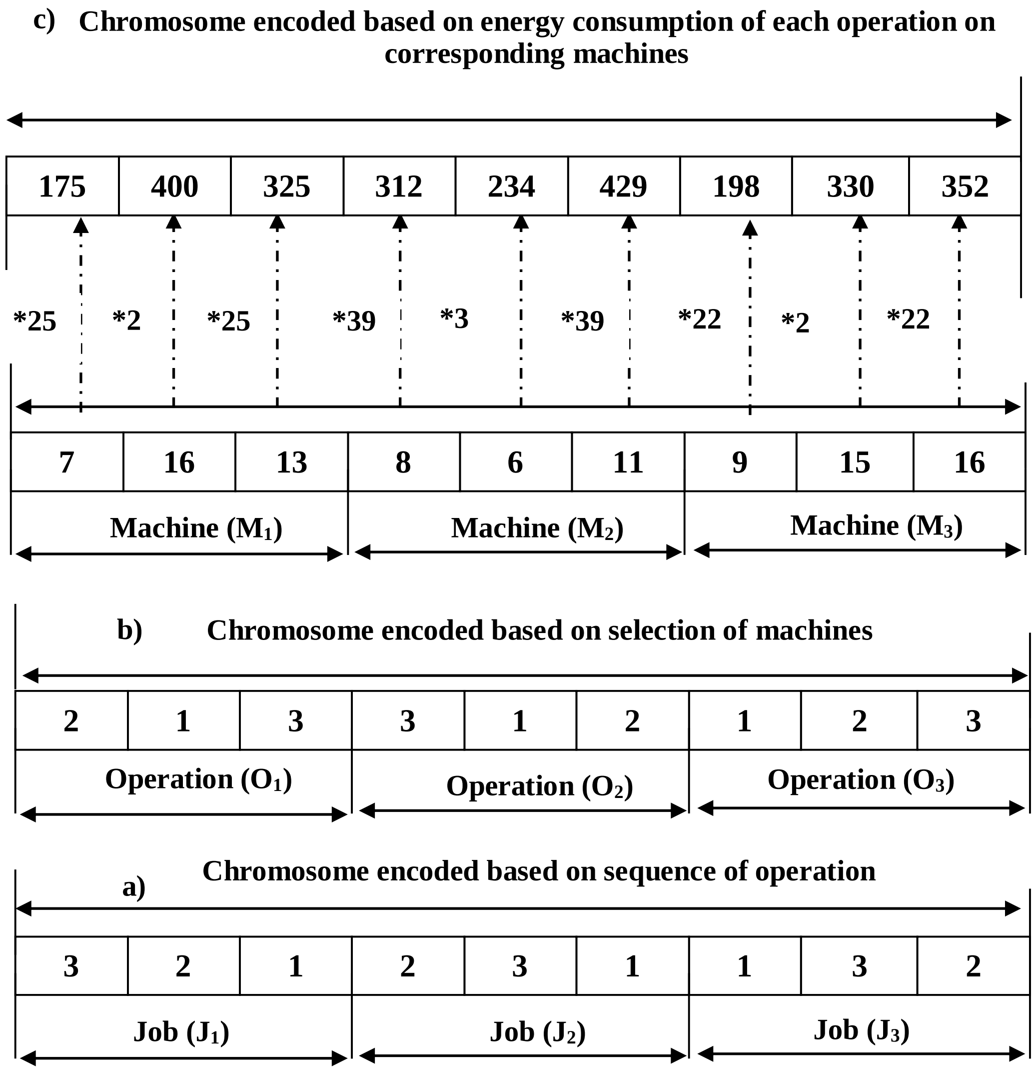


Figure 4.2. Representation of chromosome initialization for energy consumption.

The encoding schema is represented as shown in Fig. 4.2, the encoding consists of three parts. Fig 4.2 a) is encoding based on a sequence of operations of each job., which can determine the sequence of operations needed to produce a job. Fig 4.2 b) shows is encoding based on machines, which can choose the machine for each operation. Fig 4.2 c) is encoding based on the energy consumption of each machine for the corresponding operation of a particular job. Based on the encoding schema time matrices were obtained and, the time matrices for energy consumption can be represented as follows. Energy-consumption(:,:,1,2) Indicates energy consumption matrix for 2nd job 1st process plan.

,

$$
\begin{array}{c}
 \\ O_1 \\ O_2 \\ O_3 \\ O_4 \\ O_5
\end{array}
\begin{array}{c}
\begin{array}{cccccccccccc}
\mathbf{M1} & \mathbf{M2} & \mathbf{M3} & \mathbf{M4} & \mathbf{M5} & \mathbf{M6} & \mathbf{M7} & \mathbf{M8} & \mathbf{M9} & \mathbf{M10} & \mathbf{M11} & \mathbf{M12}
\end{array} \\
\left[\begin{array}{cccccccccccc}
175 & 312 & 198 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0; \\
400 & 234 & 330 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0; \\
325 & 429 & 352 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0; \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0; \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right];
\end{array}
$$

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

Step 3 A score function is defined that helps to select a suitable process plan and it is shown in Equation (4.11), where a higher score value indicates the probability of selecting the process plan is lesser. The lower the score value, the better the process plan. The formula for the score function is shown below.

$$\text{Score} = (L * E_{vk})/(R_{vk}) \tag{4.11}$$

Where L represents makespan, E_{vk} indicates energy consumption for job v on machine k, and R_{vk} indicates reliability for job v on machine k.

Step 4 A matrix CL is formed considering all the moths into the objective functions that are stored in FK represented below.

$$C = \begin{bmatrix} CL_1 & CL_2 & CL_3 & CL_4 \end{bmatrix}^T$$

A flame matrix with a similar size to the moth's matrix is considered that stores the fitness values. Even though the moth (L) matrix and flame (C) matrix consist of solutions the difference is that moths are search agents and whereas flame indicates the best position of moths

Step 5 After an appropriate process plan is selected; we consider the rows as individual light sources through which we find the minimum values.

Step 6 The matrices are explored row-wise to find a minimum entry in their respective rows once the required inputs are received and search space is clearly initialized.

Step 7 Moths maintain the best solution by updating its position Equation (4.12) by moving around the flag that is dropped by themselves during the search process. Update the position of the moth with respect to one flame. The spiral motion follows the Equation (4.12) represented as

$$Z\left(L_x, C_y\right) = S_x \cdot e^{at} \cdot \cos(2\prod t) + L_y. \tag{4.12}$$

L_x indicates the x^{th} moth, C_y indicates the y^{th} flame and Z indicates spiral function. S_x is the distance of x^{th} moth for y^{th} flame, $S_x = \| C_y - L_x \|$, a is a constant defining shape of spiral motion. Where t [-1,1],

Step 8 After finding the minimum entry in the summed matrix and converting all ∞'s to 0s, the sum of all the values is found in their respective objective function matrices.

Step 9: Finally, we will solve the function to generate optimal values for all objectives.

Table 4.3 Energy consumption data for various machines.

Machine ID	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12
Energy consumption (KJ)	15	29	32	14	11	12	19	24	14	16	22	11

The most likely solution to our problem is selected based on the most appropriate fitness value First, the data from various gear manufacturing enterprises is collected. The collected data contains the information regarding make span of the jobs, energy consumption, reliability of machines and service utilization rate. The proposed HMFO algorithm is run on the Python Jupiter Notebook on Lenovo with Intel processor with Windows 10 as OS with 64 Gigabyte RAM.

In this study, ten different instances which are real-life case data have been considered by setting the numerical quantity of jobs to be 6 and machines to be 12 where each job has a varied number of substitutable process plans. Every process plan has different set of operations. For each operation, there is a different set of machines which are capable enough to process the required task. For example, the instance 6 shown in the below Table 4.4 has 6 jobs and 12 machines in total. Job 1 has two alternative process plans and these process plans have three and three operations respectively. One operation processed on a machine at a time.

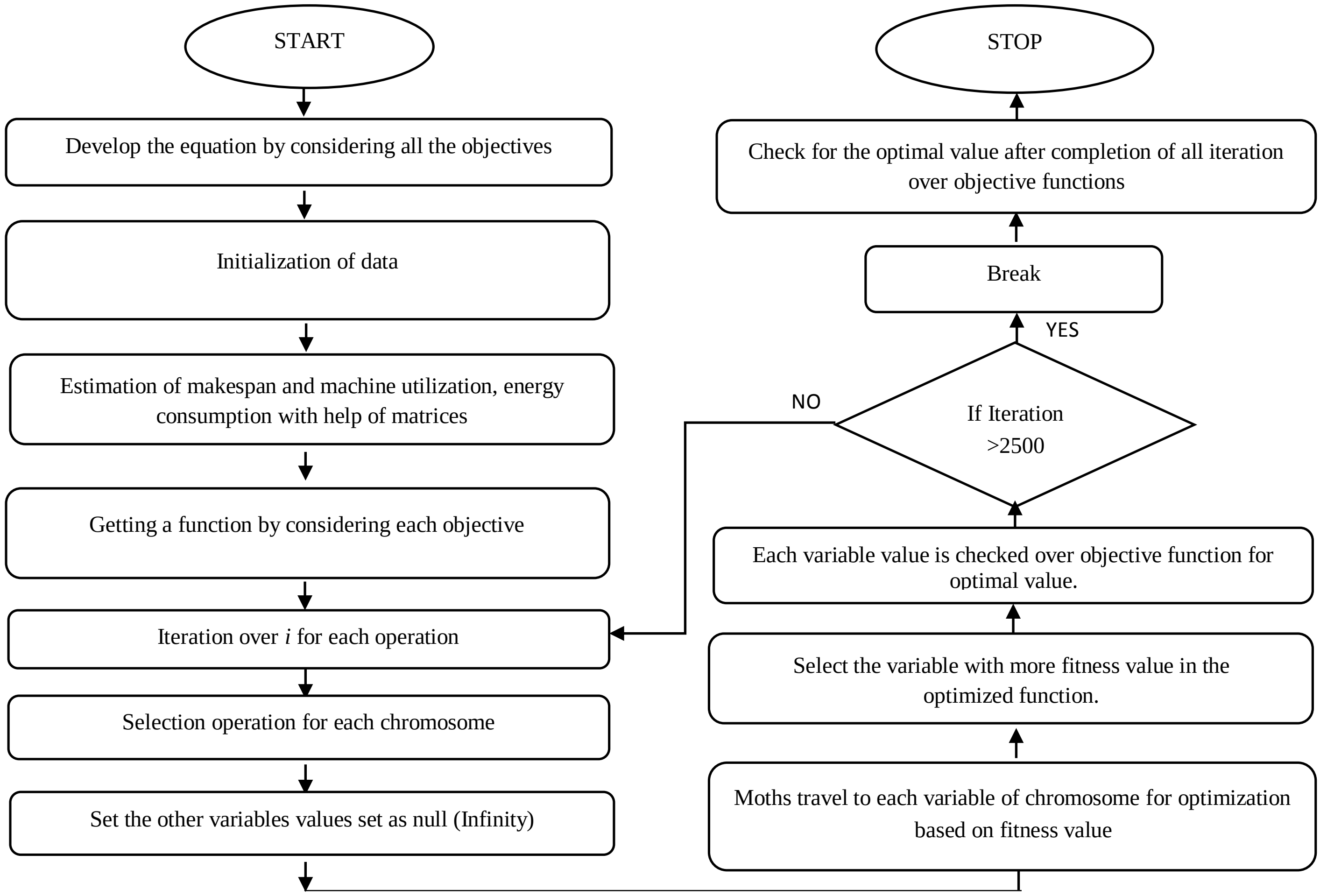


Figure 4.3 Flowchart of the proposed Hybris Moth Flame Optimization.

The processing time of various operations of jobs on several machines in different process plans, the energy consumption and reliability of all the machines is known earlier. Table 4.3 provides the energy consumed per unit time by each machine and the reliability of machine. To solve the considered problem a newly established Bio-inspired Moth Flame Optimization algorithm (HMFO) in Fig.4.3 has been adopted and further it has been mapped according to problem nature. The operations are assigned to the machines is 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.

4.4. Discussion and Results

4.4.1 Validation of proposed HMFO algorithm with the experimental instances.

To validate our approach towards optimization of makespan and energy consumption we consider some experimental instances from the literature. Table 4.4 shows the results of the experimental instances with makespan and energy consumption values. We have calculated makespan and energy consumption for around a total of 35 experiments with the data available from (Instances 1 to 32) mentioned in [99] and (Instances 33 to 35) mentioned in [100]. A comparison of our proposed HMFO results with results carried out with a Simulated Annealing Genetic algorithm (SA-GA) for instances 1 to 32. For most of the instances, the proposed HMFO gives better makespan and energy consumption values when compared with existing SA-GA makespan and energy consumption values. We also compared our proposed HMFO results with results carried out by [100], a Genetic Algorithm based Memetic Algorithm (GA-MA) for instances 33 to 35. For most of the instances, the proposed HMFO gives better makespan and energy consumption values when compared with existing GA-MA makespan and energy consumption values. All the above results indicate the better performance of proposed HMFO with some of the experimental instances which were already proposed in the literature.

Table 4.4 Results of the Experimental instances with makespan and energy consumption values.

	Jobs	Machines	Processing time range	GA-SA (Instance 1 to 32) GA- MA(Instance 33 to 35)		Proposed HMFO	
				Makespan (Minutes)	**Energy consumption (KJ)**	**Makespan (Minutes)**	**Energy consumption (KJ)**
Instance 1	3	5	[1,10]	41	138.1	**30.8**	**26.3**
Instance 2	3	7	[1,10]	54.1	205.4	**43**	**190.5**
Instance 3	3	10	[1,10]	61.2	229.1	**49.7**	**204**
Instance 4	3	5	[1,50]	190.3	708.7	**171**	**617**
Instance 5	3	7	[1,50]	252.8	960.6	**226.7**	**834**
Instance6	3	10	[1,50]	333.8	1273.3	**301.7**	**1110**

Instance7	3	5	[1,100]	375.4	1307.1	**319.3**	**1134**
Instance8	3	7	[1,100]	531.9	1895.4	**516.7**	**1644.9**
Instance9	3	10	[1,100]	729.1	2830.5	**698**	**2467.5**
Instance10	5	5	[1,10]	35	140.4	**24**	**126.1**
Instance11	5	7	[1,10]	46	186.3	**30**	**169**
Instance12	5	10	[1,10]	51.5	199.9	**43.7**	**177**
Instance13	5	5	[1,50]	165.5	671.5	**149.8**	**583.9**
Instance14	5	7	[1,50]	225.2	951.2	**201.7**	**828**
Instance15	5	10	[1,50]	317	1303.6	**306.8**	**1139**
Instance16	5	5	[1,100]	325.5	1253.2	**311.7**	**1098**
Instance17	5	7	[1,100]	436.9	1909	**410**	**1663**
Instance18	5	10	[1,100]	610.3	2587.5	**598**	**2257**
Instance19	7	5	[1,10]	28.7	110.9	**19.6**	**96.7**
Instance20	7	7	[1,10]	39.3	162.2	**24**	**146**
Instance21	7	10	[1,10]	56.5	241.6	**49**	**218**
Instance22	7	5	[1,50]	159.7	607	**143**	**524**
Instance23	7	7	[1,50]	220.8	919.1	**206**	**846**
Instance24	7	10	[1,50]	304.6	1310.5	**265.002**	**1150.9**
Instance25	7	5	[1,100]	351	1422.9	**305.37**	**1287.9**
Instance26	7	7	[1,100]	426.1	1978.3	**370.7**	**1725**
Instance27	7	10	[1,100]	625.9	2664.1	**544.5**	**2319.7**
Instance28	10	10	[1,200]	939.04	9873.2	**816.96**	**8597.6**
Instance29	15	15	[1,200]	1554.12	22505.2	**1352.08**	**19579.5**
Instance30	20	20	[1,200]	4778.07	80577.2	**4156**	**70102.16**
Instance31	20	20	[1,200]	7753.04	100073.4	**6749**	**87263.8**
Instance32	20	20	[1,200]	15062.5	197787.5	**13115**	**172975.7**
Instance 33	18	15	[1,200]	531	13340.3	**502**	**12986**
Instance 34	18	15	[1,200]	810	2036.32	**739**	**1956**
Instance 35	18	15	[1,200]	680	2267.88	**593**	**1837.6**

4.4.2. Evolution of Proposed HMFO with Practical instances

After proving the effectiveness of the proposed HMFO with practical instances in Table 4.5. Furthermore, the effectiveness of the proposed algorithm is tested on different problems of practical instances with the aim to minimize makespan and energy consumption, maximizing machine utilization and reliability. Table 4.5 describes the optimal process plans chosen for each job in all the instances. Out of the various alternative process plans, only one process plan per job is chosen depending upon the score value. The lower the score value, the better the process plan. So, whichever process plan gives the lowest score value, that process plan is selected. For example, in instance 6 the process plans selected for the jobs 1-6 are 1, 1, 1, 1, 2 and 1 respectively.

Table 4.5 Optimal process plans selected for each job for all practical Instances 1 to 10

	Case		**Process plans selected**							
	Jobs	**Machines**	**J1**	**J2**	**J3**	**J4**	**J5**	**J6**	**J7**	**J8**
Instance1	6	6	2	2	2	2	2	2	-	-
Instance2	6	6	1	1	2	2	1	2	-	-
Instance3	6	8	3	1	2	3	3	4	-	-
Instance4	8	8	2	2	2	2	1	2	1	3
Instance5	8	8	2	1	2	1	1	1	2	2
Instance6	6	12	1	1	1	1	2	1	-	-
Instance7	6	12	2	2	2	3	2	2	-	-
Instance8	6	12	2	2	2	2	2	2	-	-
Instance9	6	12	2	1	1	2	2	1	-	-
Instance10	6	12	2	1	1	3	2	1	-	-

Table 4.6 shows the optimal values of makespan and energy consumption for all different problem instances (1 to 10) of HMFO and NSGA-II. These are the Pareto optimal values obtained by the simultaneous optimization of all the objectives. For example, in instance1, for proposed HMFO the Pareto optimal value of makespan is 30 and energy consumption is 8906 and with NSGA-II the values of makespan and energy consumption as 43 and 9083 respectively. And for instance6, for proposed HMFO the value of makespan is 986 and energy

consumption is 15553 and with NSGA-II we got the values of makespan and energy consumption as 1083 and 17694 respectively.

Table 4.6 Results of the practical instances with makespan and energy consumption values

	Jobs	Machines	Proposed HMFO		NSGA-II	
			Makespan (Minutes)	Energy consumption (KJ)	Makespan (Minutes)	Energy consumption (KJ)
Instance 1	6	6	30	8906	35	9083
Instance 2	6	6	27	8325	38	8700
Instance 3	6	8	179	39658	187	44920
Instance 4	8	8	42	13089	50	14040
Instance 5	8	8	50	10129	62	11189
Instance6	6	12	986	15553	1083	17694
Instance7	6	12	1179	15224.5	1256	16785
Instance8	6	12	1026	13881	1094	14205.5
Instance9	6	12	669	14298	756	14898.5
Instance10	6	12	814	14143.5	884	14678

To validate the performence of the proposed algorithm, ten different instances are considered and their considered objectives makespan and, energy consumption values are shown in Table 4.6. From the Table 4.6 clear observation indicates Pareto optimal values of makespan and energy consumption values are dependent on the number of jobs and machines. Interestingly makespan and energy consumption values for Instance 3 i.e. six jobs and eight machines (6*8) case is more when compared to instances other instances 1,2 i.e. six jobs, six machine case (6*6) and Instances 4, 5 i.e. eight jobs, eight machine case (8*8). A similar trend is also shown in the literature for 6*8 cases for makespan and energy consumption values. In almost all instances, the make span and energy consumption values of HMFO are lesser and far better than those given by NSGA-II. Thus, on comparing both the algorithms we can conclude that the optimum values are attained in the case of HMFO which proves the effectiveness of the proposed multi-objective evolutionary algorithm.

For Instances 6 to 10, Even though there are equal instances of jobs and machines, the make span and energy consumption values are different and depend only on the operations and types of machines used.

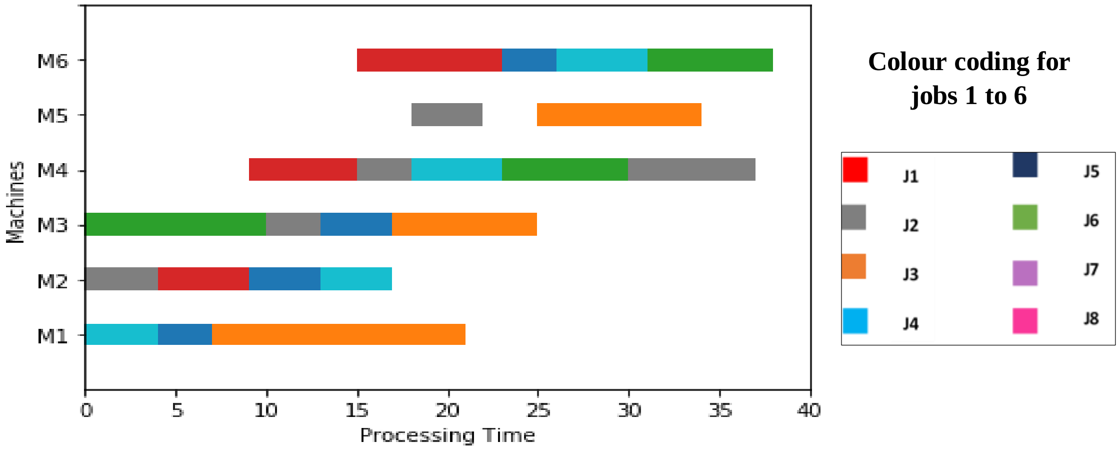


Figure 4.4 Gantt chart showing the make span of instance 1.

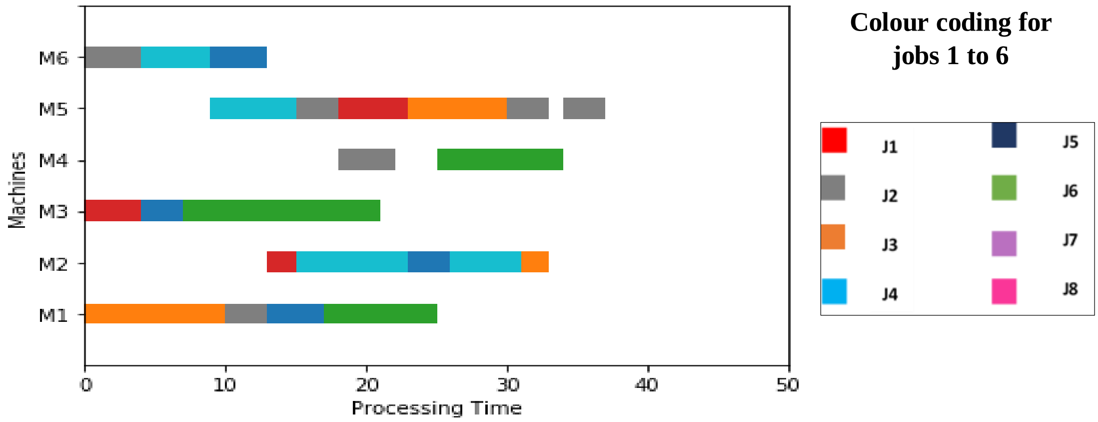


Figure 4.5 Gantt chart showing the makespan of instance 2.

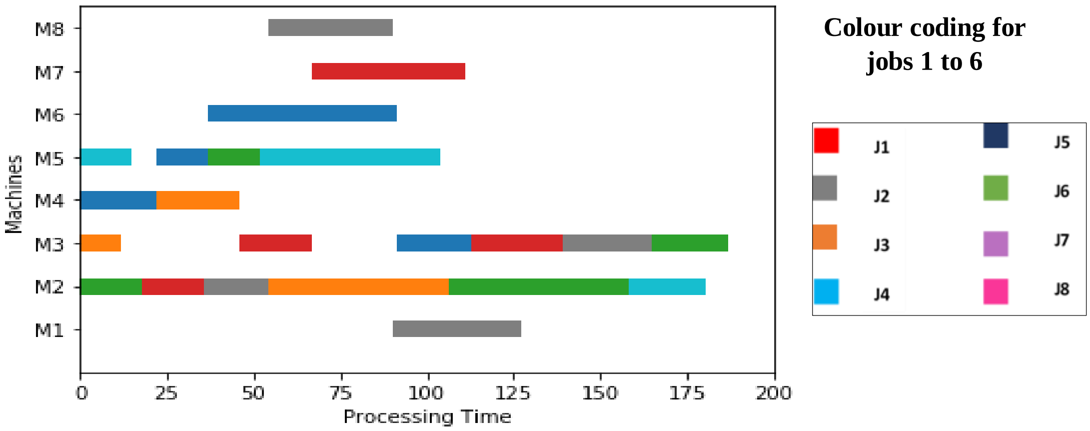


Figure 4.6 Gantt chart showing the make span of instance 3.

For a better portrayal of the optimized HMFO algorithm's results, Gantt charts have been utilized for all ten instances. For better understanding Gantt charts for Instances, 1 to 5 are shown separately in Fig. 4.4, Fig. 4.5, Fig. 4.6, Fig. 4.7, Fig. 4.8 and Gantt charts for Instances 6 to 10 are shown separately in Fig. 4.9, Fig. 4.10, Fig. 4.11, Fig. 4.12 , Fig. 4.13. Here in Fig. 4.4 to Fig. 4.13 illustrate the maximum completion time for the problem with respect to instances 1 to 10 respectively.

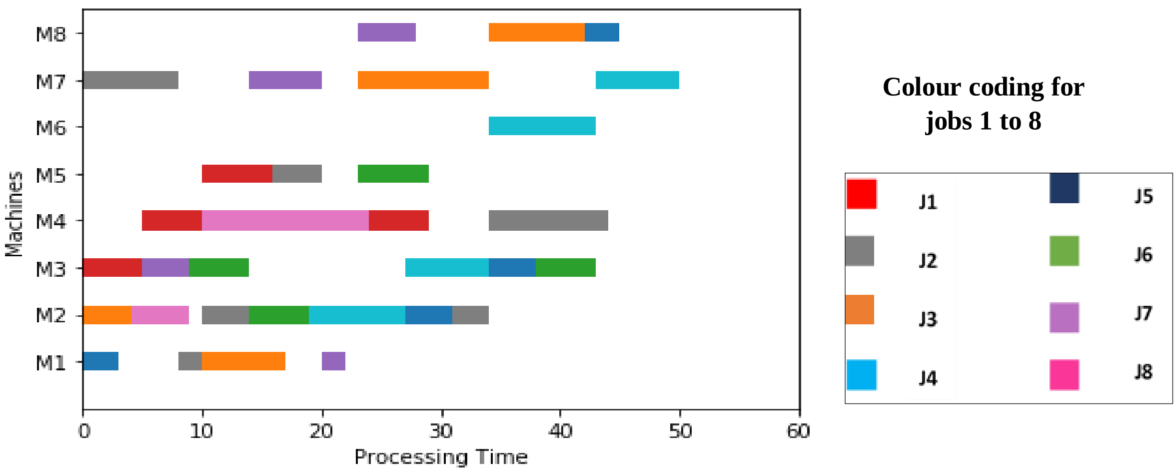


Figure 4.7 Gantt chart showing the make span of instance 4

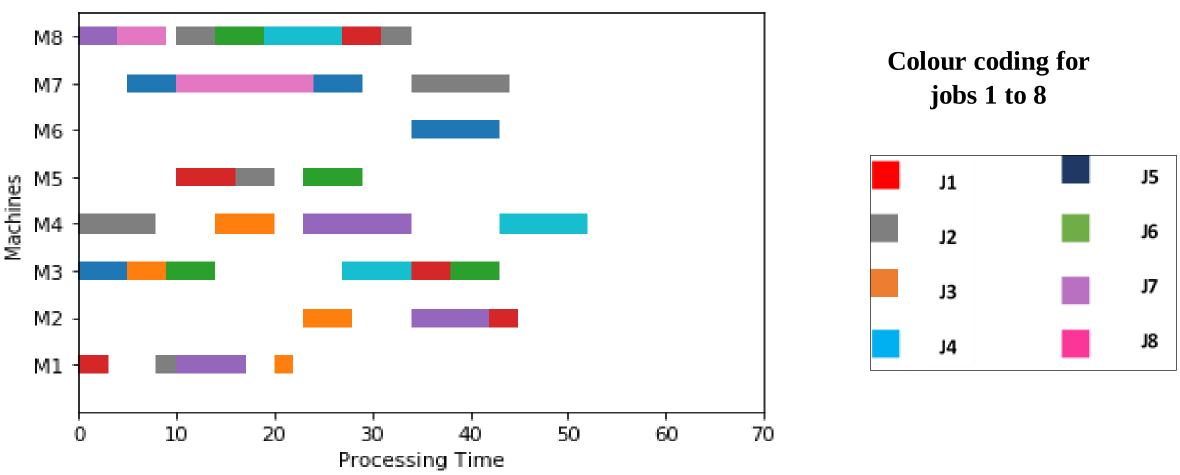


Figure 4.8 Gantt chart showing the make span of instance 5

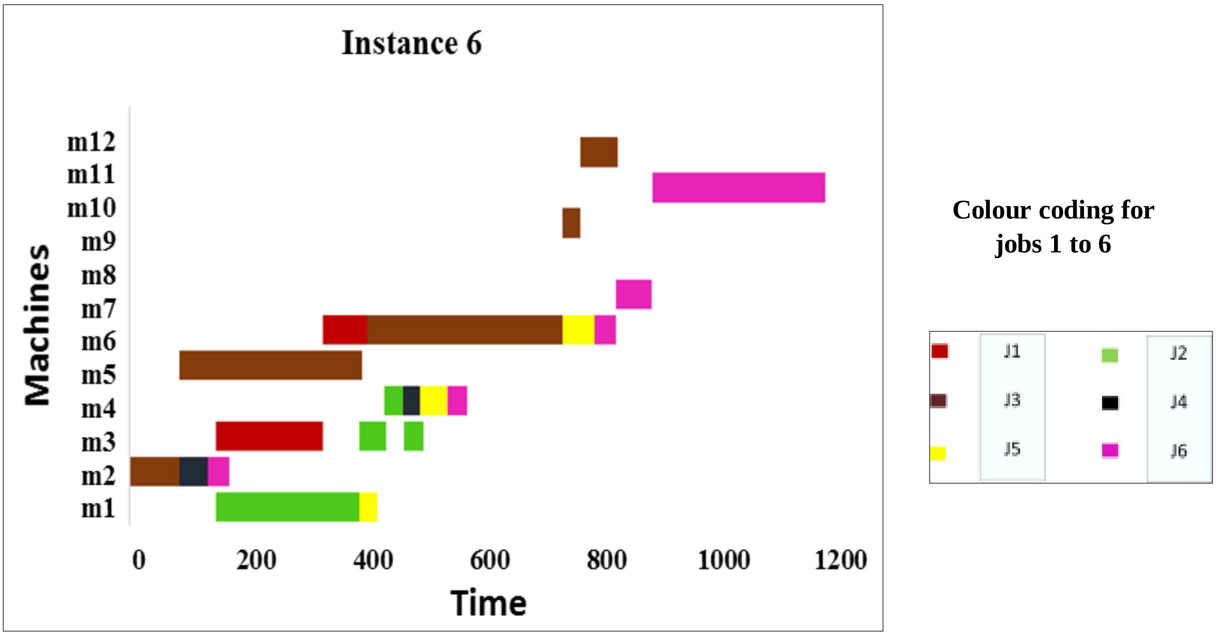


Figure 4.9 Gantt chart showing the make span of instance6

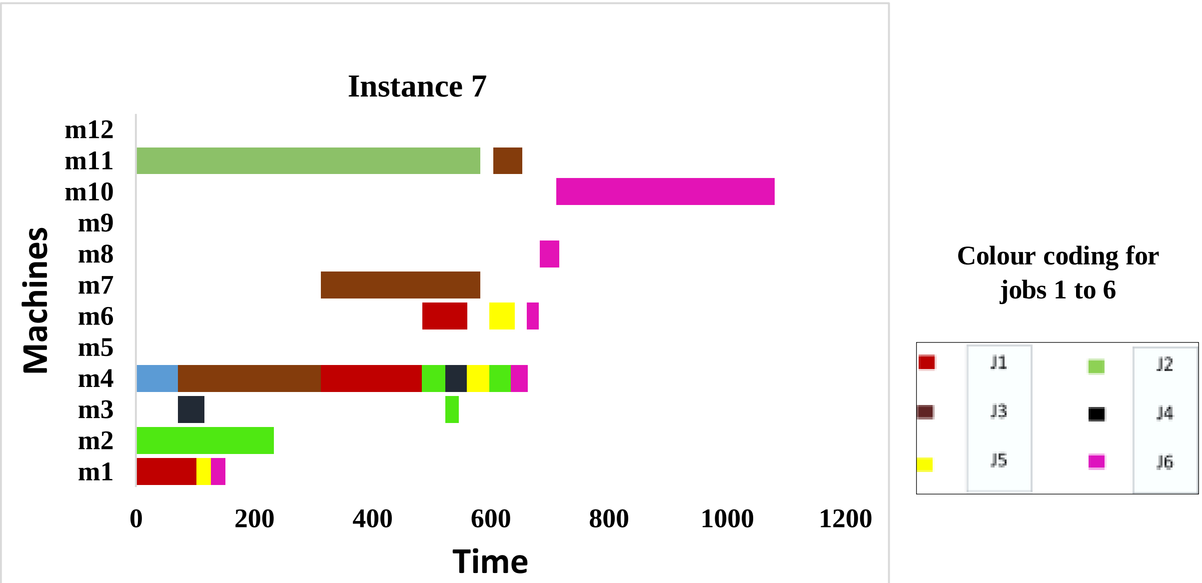


Figure 4.10 Gantt chart showing the make span of instance 7.

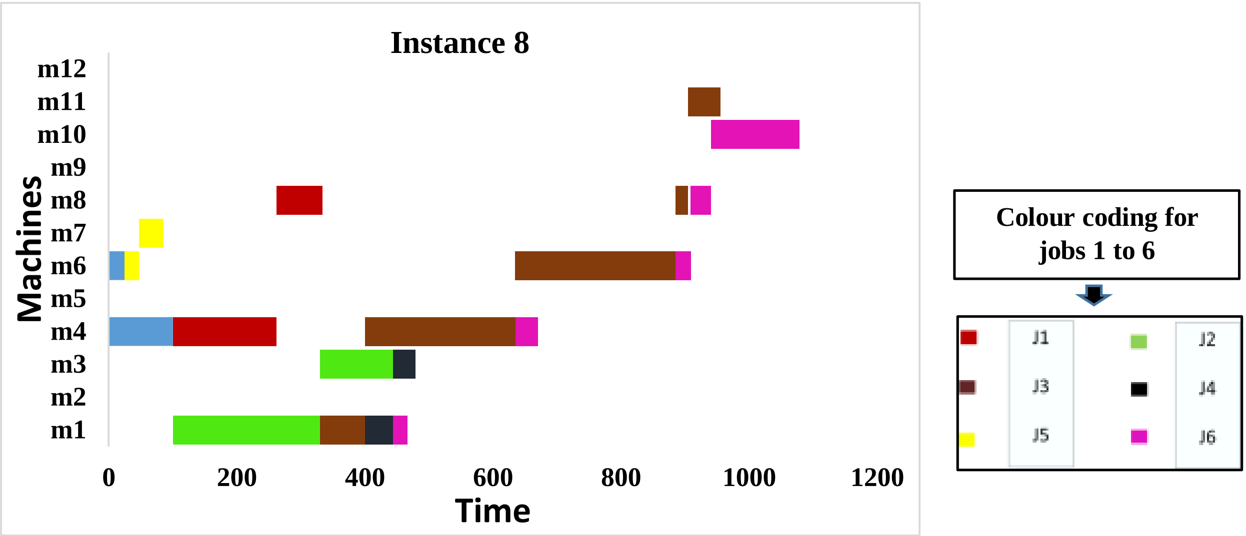


Figure 4.11 Gantt chart showing the make span of instance 8.

The X-axis of the Gantt chart indicates the average time of completion of the job (makespan) and Y-axis denotes the machines. As stipulated in Fig 4.4 to Fig. 4.8, it is clearly observable that the makespan for all the five instances is 30,27,179,42, and 50 respectively., which replicates the results that are previously mentioned in Table 4.6 From the Fig 4.9 to Fig. 4.13, it is clearly observable that the makespan for all the five instances is 986, 1179, 1026, 669 and 814 respectively, which replicates the results that are previously mentioned in Table 4.6.

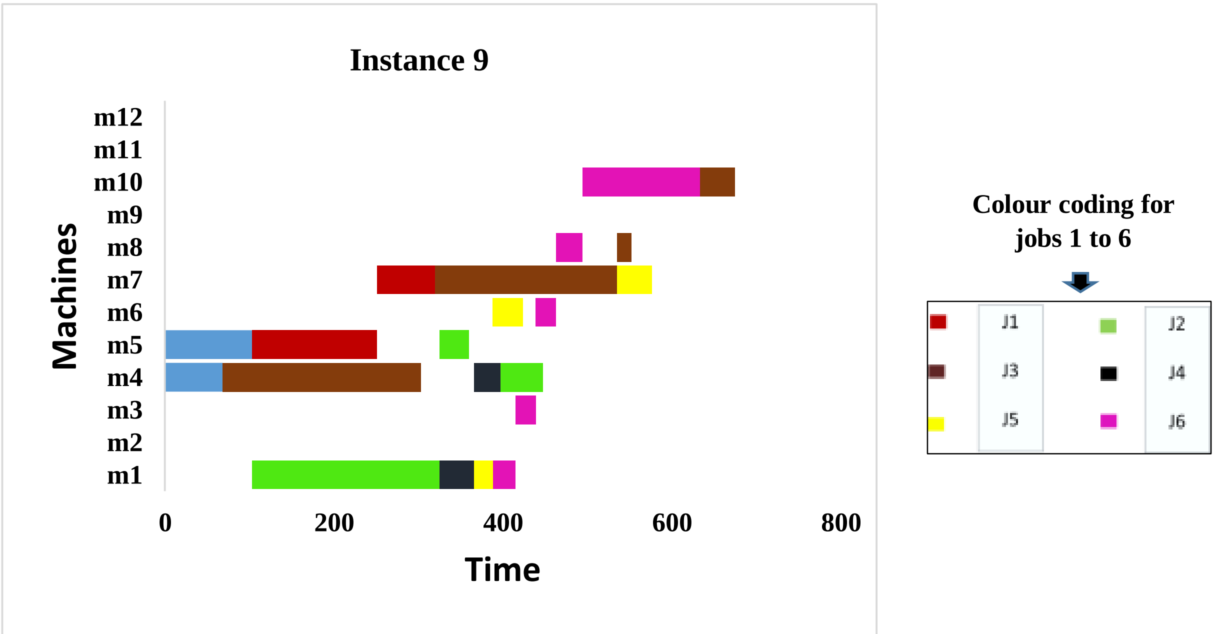


Figure 4.12 Gantt chart showing the make span of instance 9.

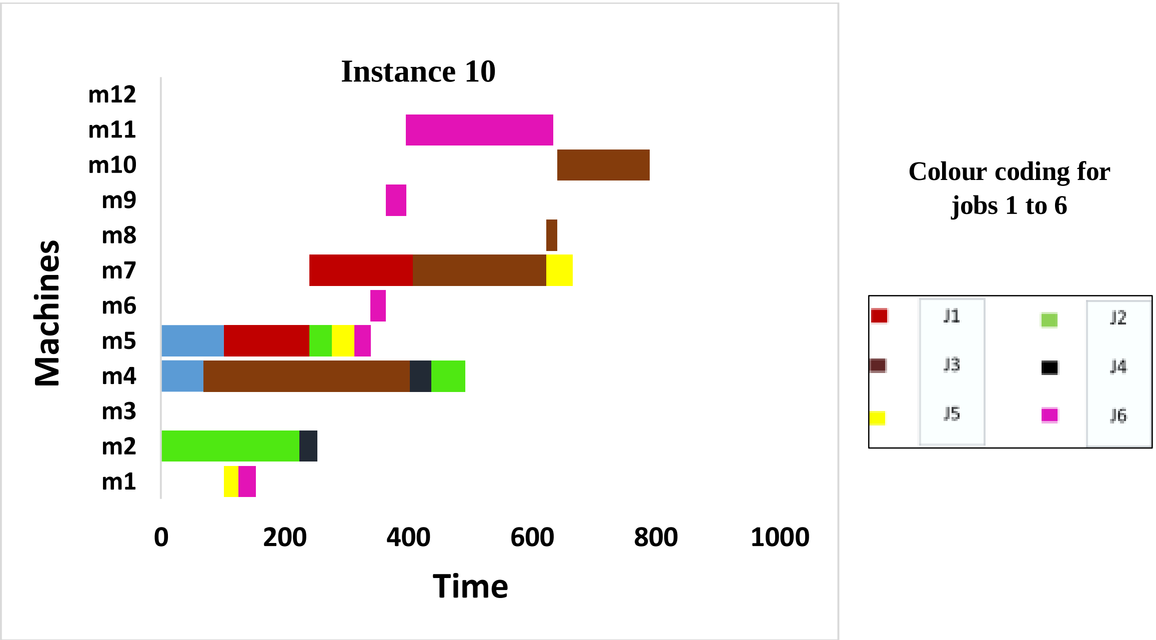


Figure 4.13 Gantt chart showing the make span of instance 10

In addition, to comparing the performance of both the algorithms i.e., HMFO and NSGA-II a comparative study of the machine utilization rate of different machines for all the ten instances is illustrated in Fig. 4.14, Fig. 4.15, Fig. 4.16, Fig. 4.17, Fig. 4.18, Fig. 4.19, Fig. 4.20, Fig. 4.21, Fig. 4.22, Fig. 4.23 respectively.

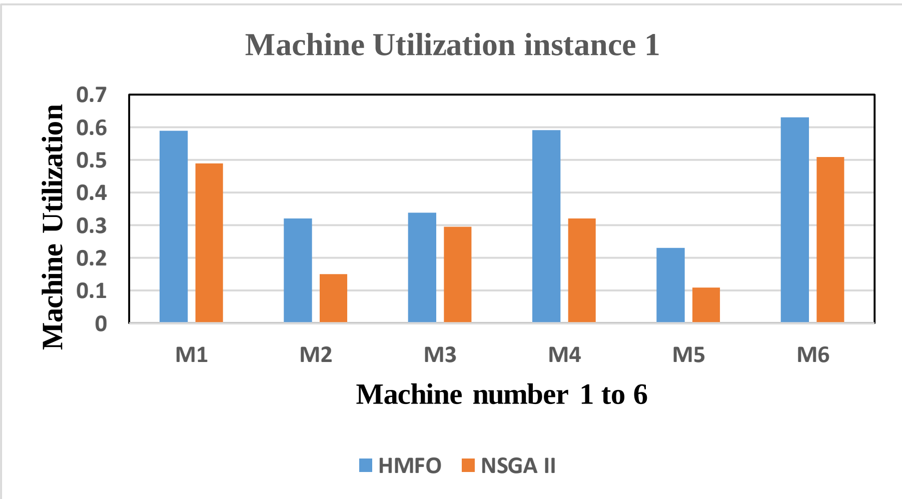


Figure 4.14 Machine Utilization values for instances 1

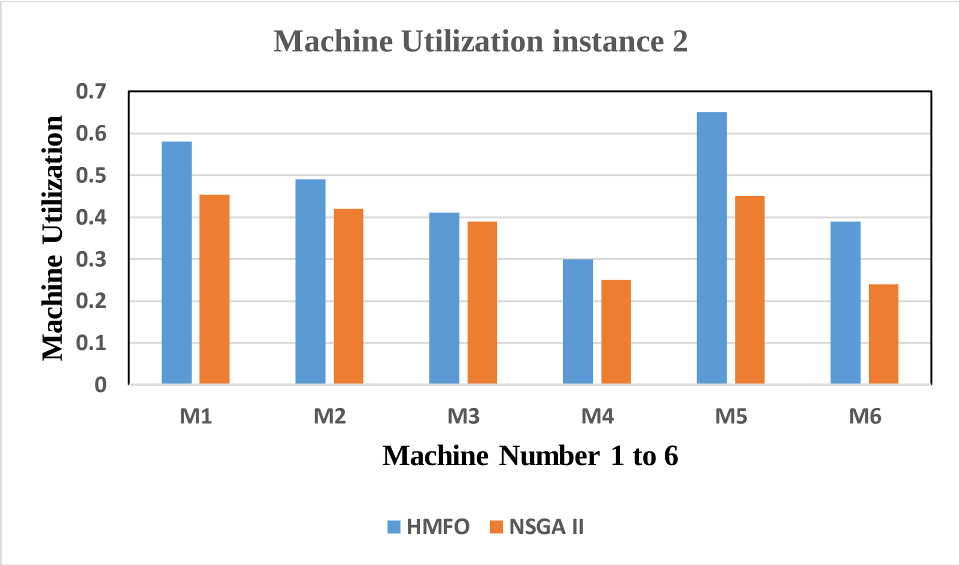


Figure 4.15 Machine Utilization values for instance 2

It can be inferred from Fig.4.14 to Fig. 4.18 all instances 1 to 5 machine utilization rates are far better for the results that are obtained with HMFO when compared with results that are obtained with NSGA II. From Fig. 4.19 for Instance 6 in case of HMFO, Machine 6 has the maximum utilization rate and machine 9 has the minimum utilization rate and in case of NSGA-II, machine 6 has the highest and machine 12 has the least utilization rate.

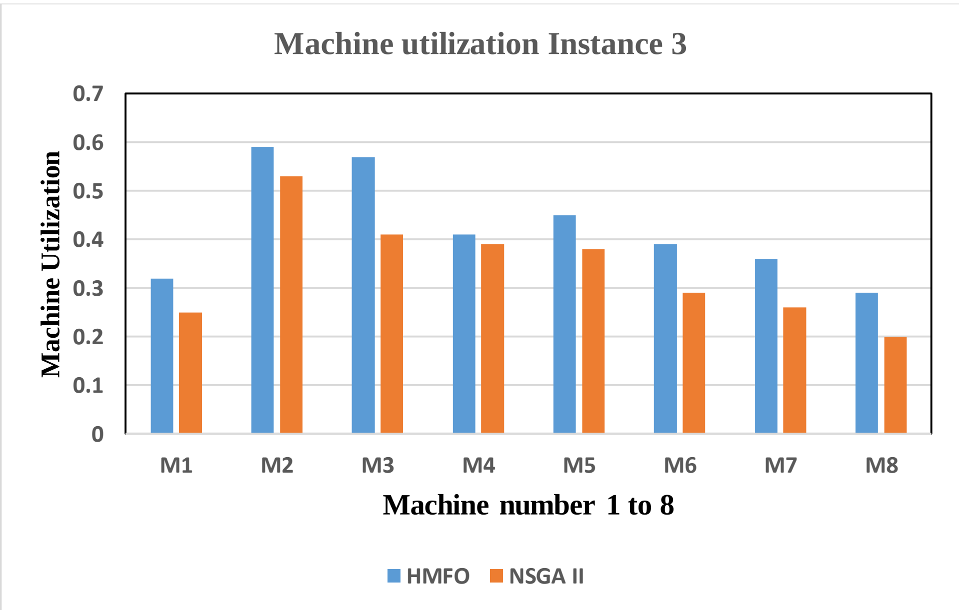


Figure 4.16 Machine Utilization values for instance 3

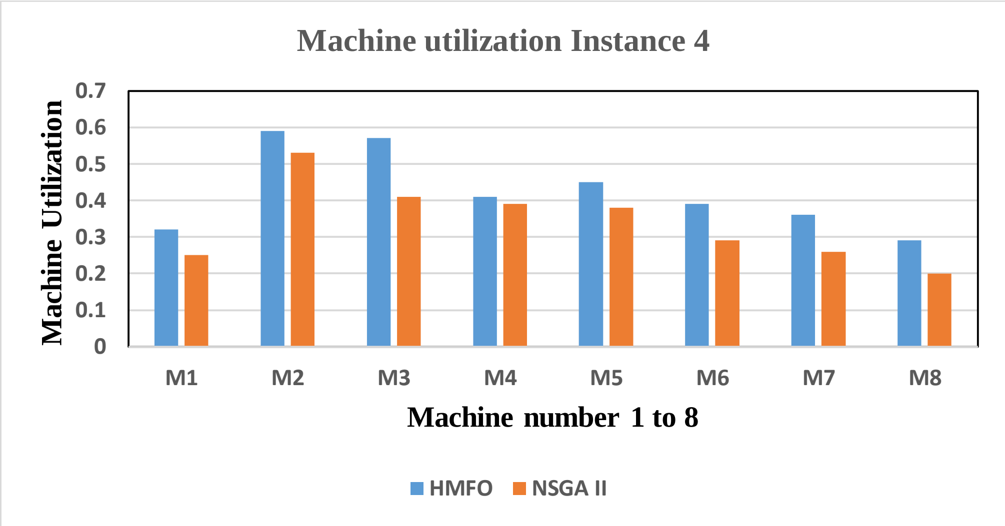


Figure 4.17 Machine Utilization values for instances from 4

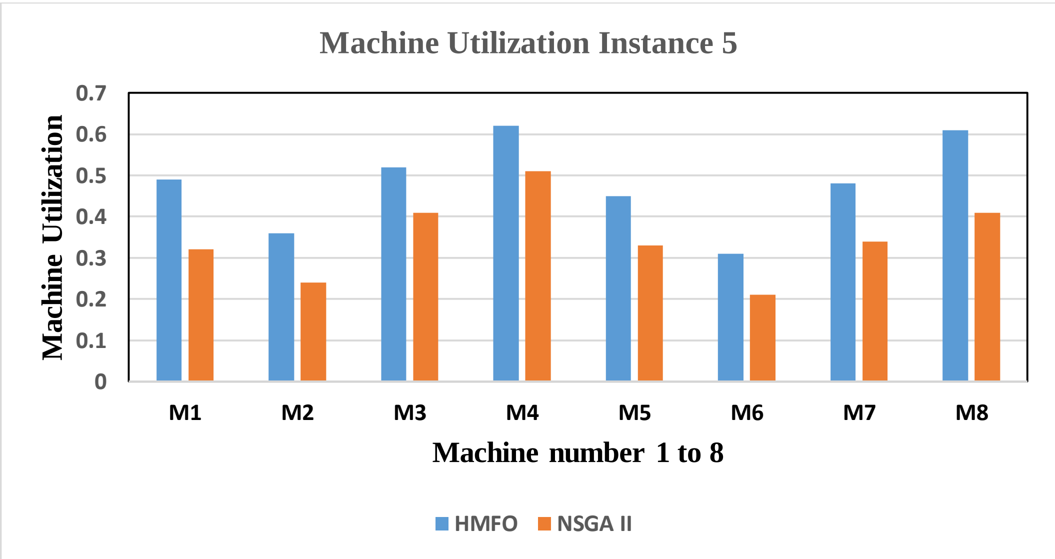


Figure 4.18 Machine Utilization values for instances 5

Also, it is evident from the bar graph that in the case of HMFO, machine 8 and machine 12 have zero utilization rate, it means that these machines are not at all utilized in the manufacturing process and in case of NSGA-II all the machines are completely utilized. Since our objective is the maximization of service utilization rate, the machines that are not utilized can be completely removed from ignored and can be removed from the workspace. Since

HMFO reduces the initial cost involved in installing the machines by eliminating the machines. We can conclude that HMFO gives the optimum results of the machine utilization rates.

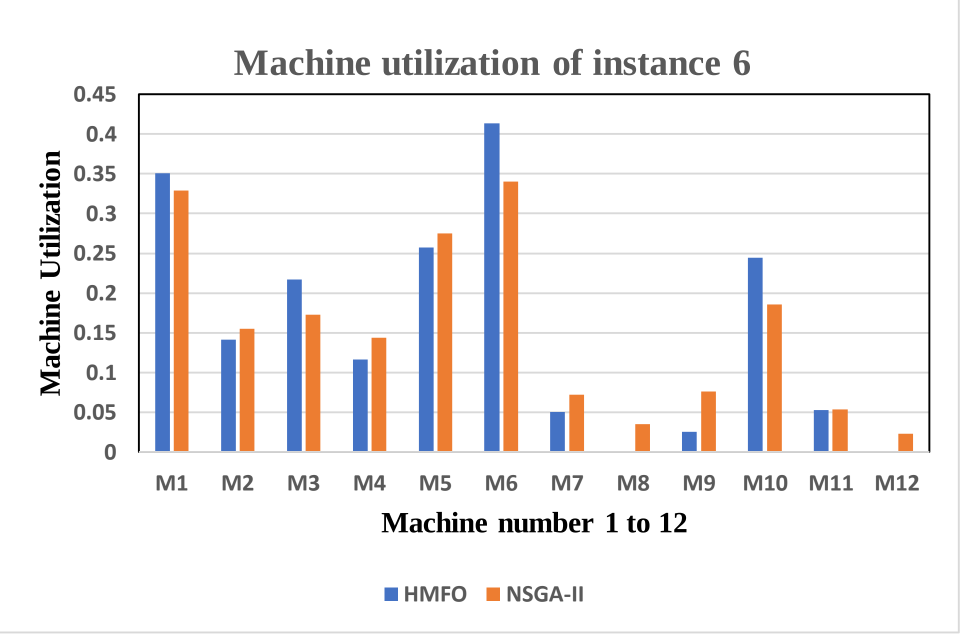


Figure 4.19 Machine Utilization values for instances from 6

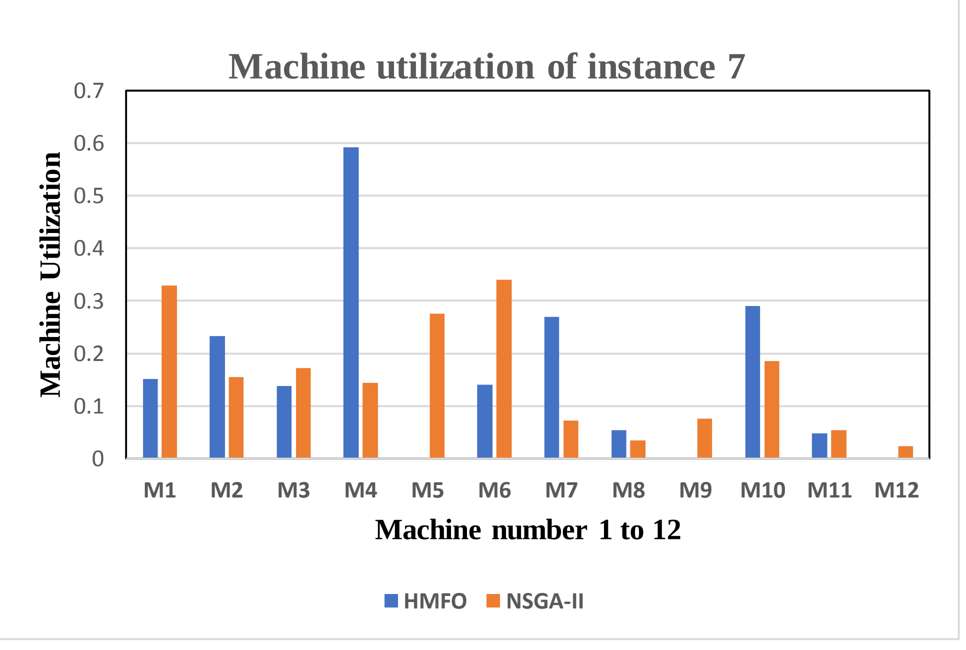


Figure 4.20 Utilization rate of different machines of instance 7

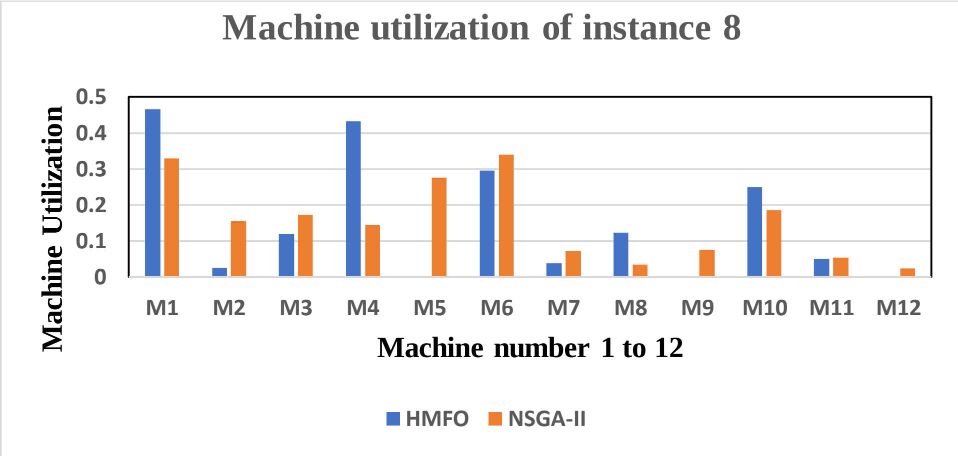


Figure 4.21 Utilization rate of different machines of instance 8

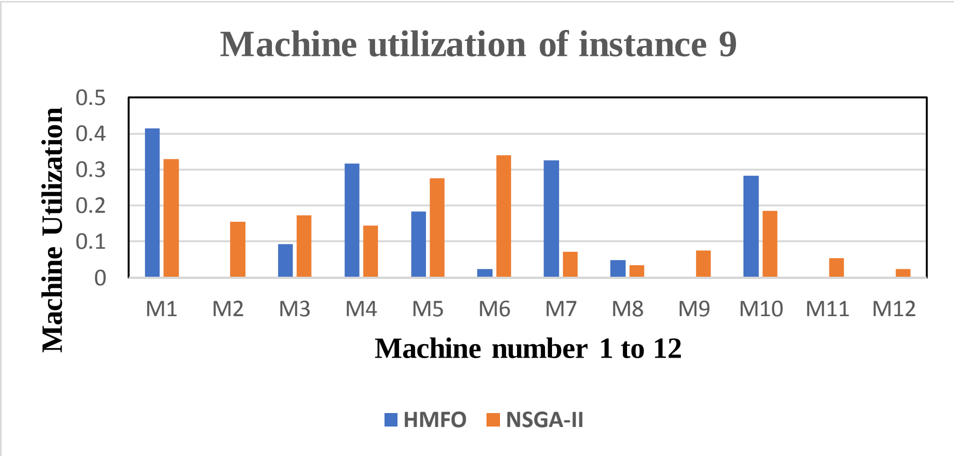


Figure 4.22 Utilization rate of different machines of instance 9.

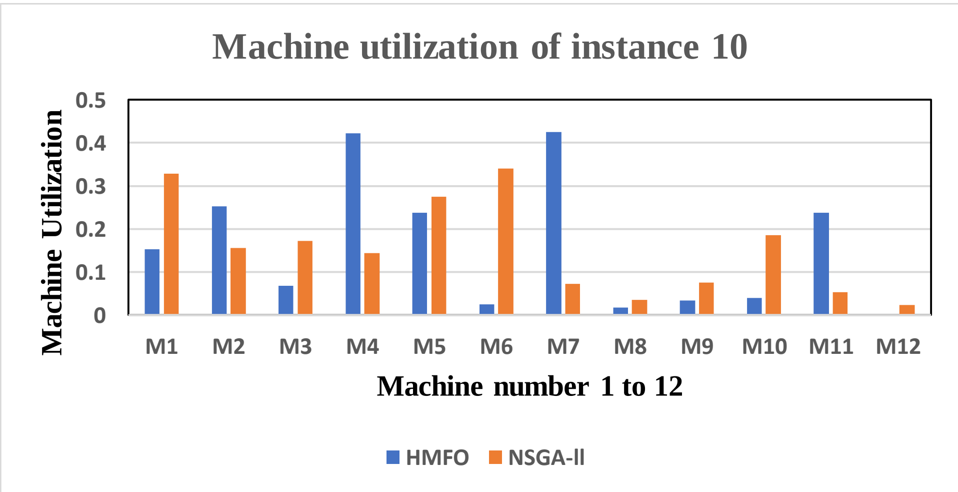


Figure 4.23 Utilization rate of different machines of instance 10

From Table 4.6 the energy consumption values for all instances are plotted and shown in the Fig. 4.24. It can be inferred that Energy consumption values for all ten instances are lower for HMFO when compared to NSGA II.

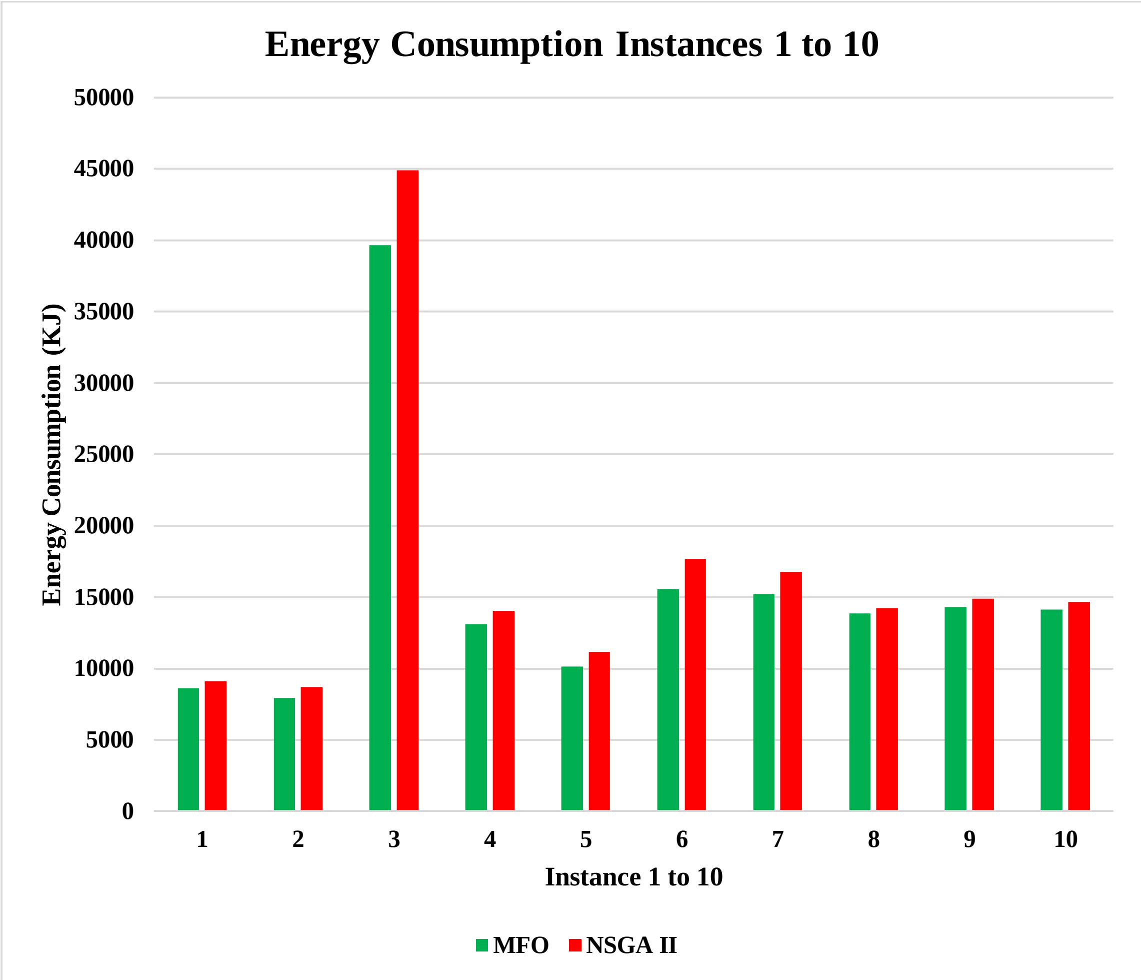


Figure. 4.24 Energy consumption values of all instances 1 to 10

From the above the energy consumption values are reduced drastically with the proposed algorithm that indicates better performance of the algorithm. The results are compared with HMFO with NSGA II. All the considered ten instances are shown in the Figure 4.24. Moreover, the energy consumption values are very high for the Instance 3 when compared to all other instances. From the instances 6 to 10 the energy consumption values are maintained uniformly but the proposed HMFO consume lesser energy when compared to the NSGA II.

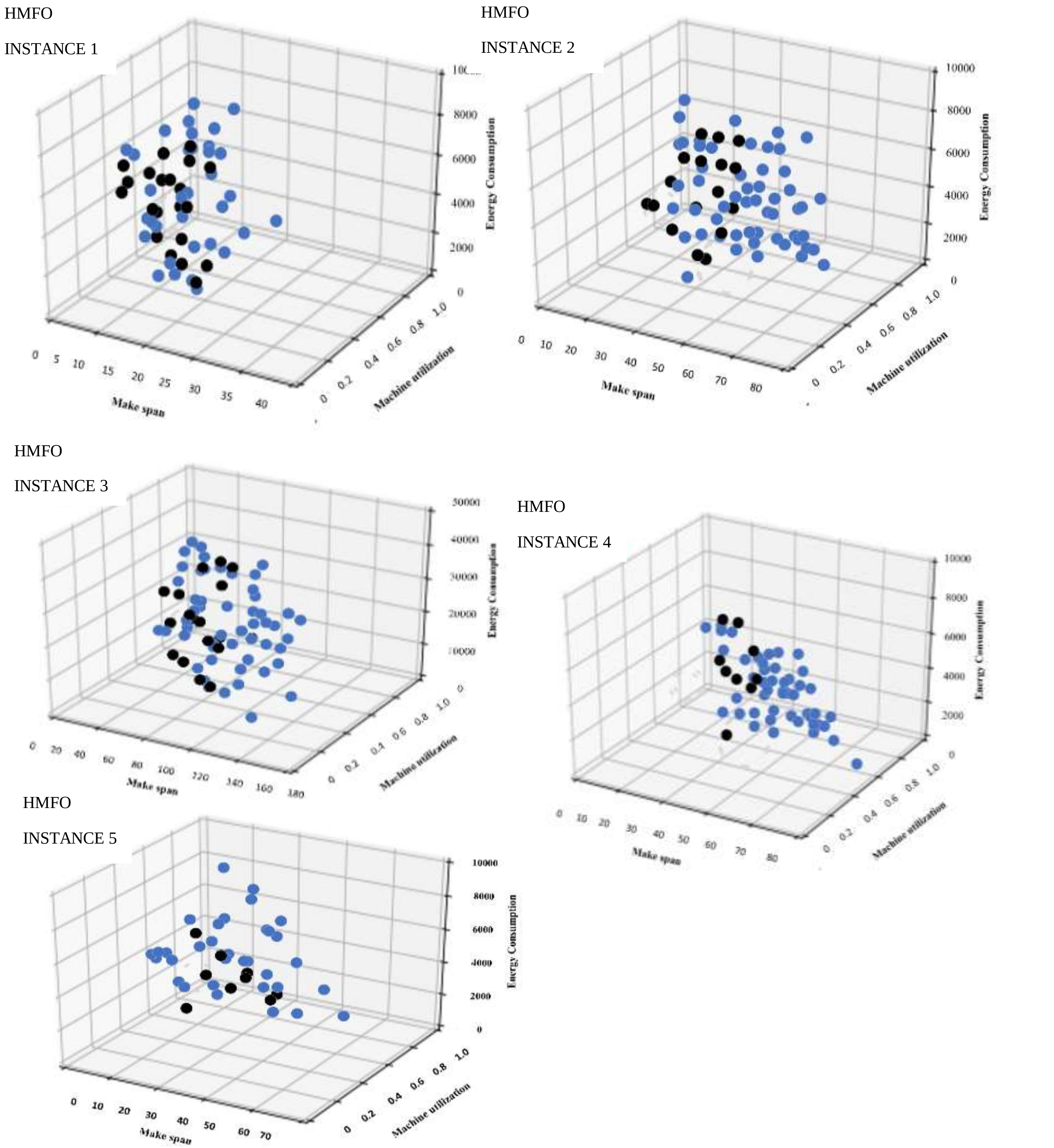


Figure. 4.25 Pareto optimal graphs showing various solutions for three objectives makespan, energy consumption and, Machine utilization for HMFO algorithm.

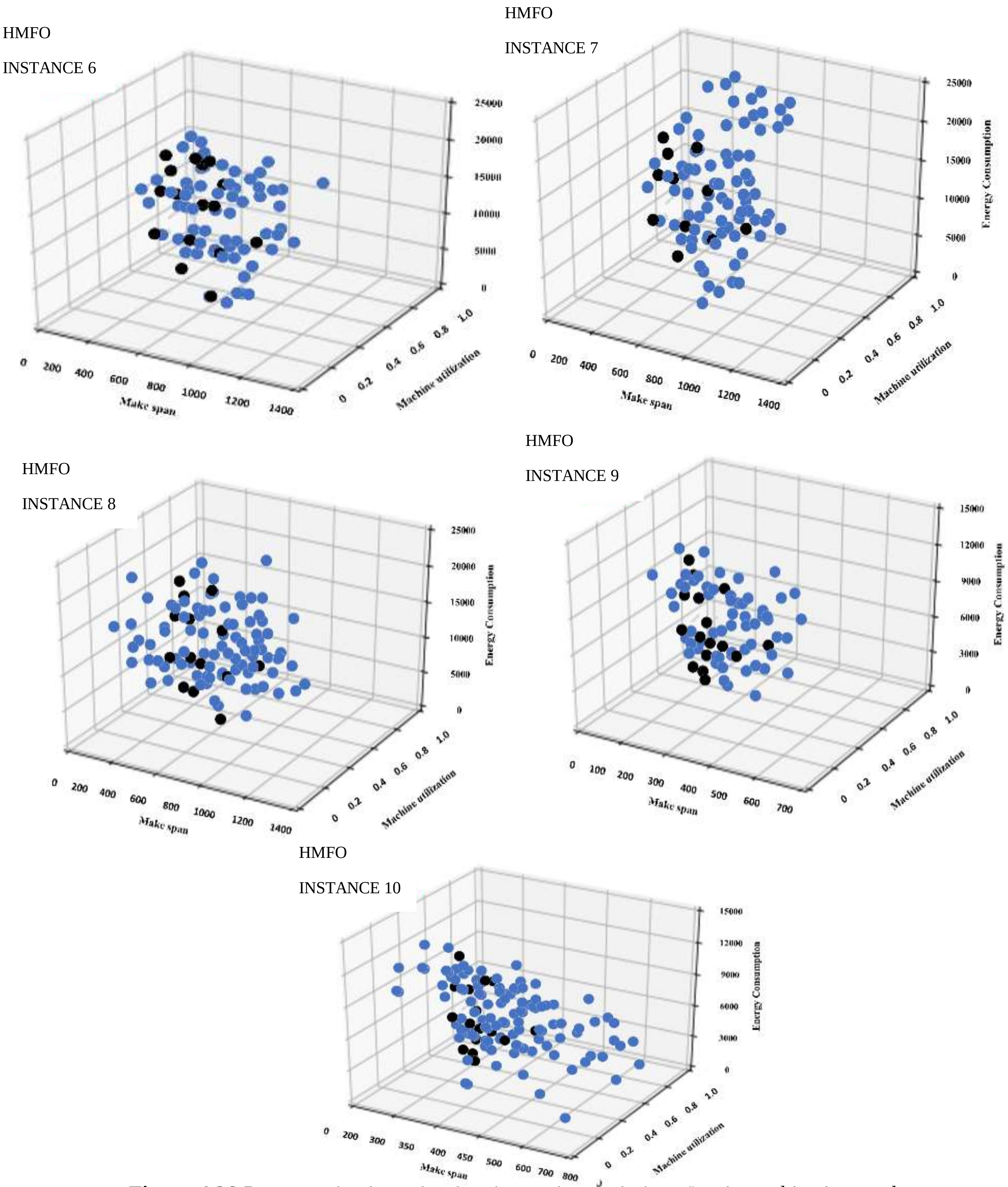


Figure 4.26 Pareto optimal graphs showing various solutions for three objectives makespan, energy consumption and, Machine utilization for HMFO algorithm.

4.5. Performance Indicators (PI) for the algorithms

To make the comparison of proposed algorithm with the benchmark algorithms several performance measures were reported [101] [102] [103] [104] and these measures mainly useful for multi/many objective optimization problems. To make the process simple and effective Pareto optimal graphs were plotted for three objectives shown in Fig.4.25 and Fig. 4.26. From Fig. 4.25 and Fig 4.26 gives various Pareto graphs generated by HMFO and NSGA –II respectively. In Fig 4.25 and Fig. 4.26 the black coloured solutions indicating the non-dominated solutions and blue colour solutions indicating the remaining solutions. To get the effective evolution of the algorithms performance considering all of them will give a better picture.

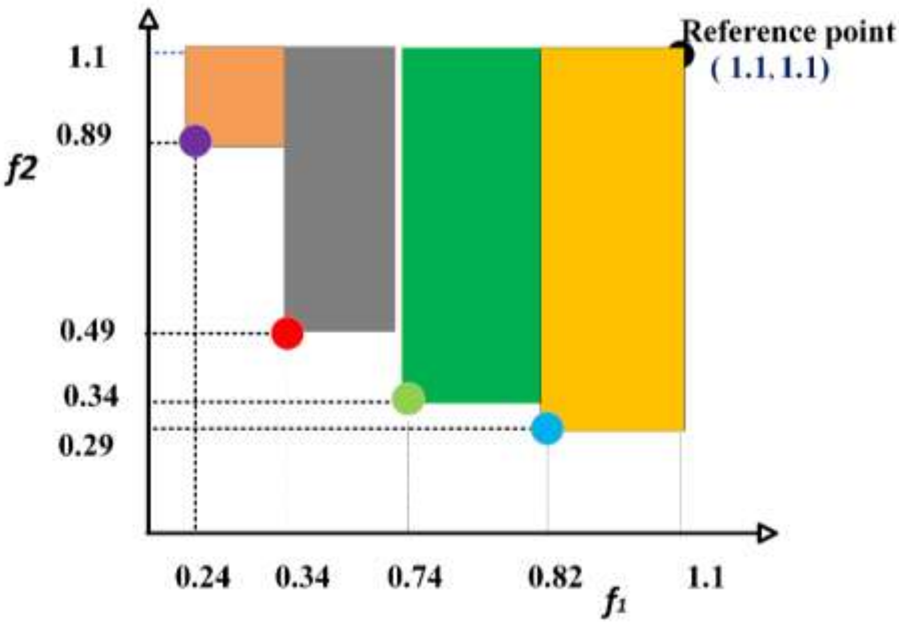


Figure.4.27 Model hypervolume calculation for better understand.

The hypervolume (HV) is most used Performance Indicator (PI) among all for the comparison multi objective algorithms [56]. HV is the volume occupied by the dominated Pareto front approximation ‘R’ drawn from a reference point o £ O^L, such that Y £ R, R<o. The HV is given by Equation (4.13). Here η_L represents L dimensional lebesgue measure.

$$HV(R,\ o) = \eta_L(\bigcup_{Y \in R} [Y, o] \qquad (4.13)$$

The HV describes the region of the objective space which will be weakly dominated by the approximation set. Till now there are no particular guidelines for selecting the reference point, however the worst possible point (i.e dominated by all points) then the nadir point (1, 1, …1) will be considered in most of the studies shown in Fig. 4.27. [105]. Here the HV is calculated by specifying the reference point as (1.1, 1.1). Out of all the available HV values, the Best, median, least values of hypervolume (HV) results are indicated with help of Box-plot in Fig. 4.12. The HV is calculated for various instances of problems of proposed HMFO and NSGA-II algorithms represented in Fig. 4.28. For Instance- 1 in the case of HMFO the best, median, least values are 0.6997, 0.6297, 0.5497, and similarly for Instance-1 in the case of NSGA II the best, median, least HV values are 0.5653, 0.4853, 0.3653 respectively.

In a similar manner, the box plots were obtained for all ten instances that are represented in Fig.4.28. It is well known from the literature the Higher the HV value better is the performance of the algorithm. From the after thorough observation of HV values indicating the higher values for all the instances (i.e Instance 1 to Instance 10) in case of proposed HMFO when compared to the NSGA –II algorithm. This demonstrates the superiority of the proposed HMFO over the NSGA II algorithm for better approximation. Moreover, HV results for the first five instances (Instances 1 to Instance 5) are falling in a lower range when compared to the other instances (Instances 6 to Instance 10). This may be due to different problem scenarios that were considered in all ten instances.

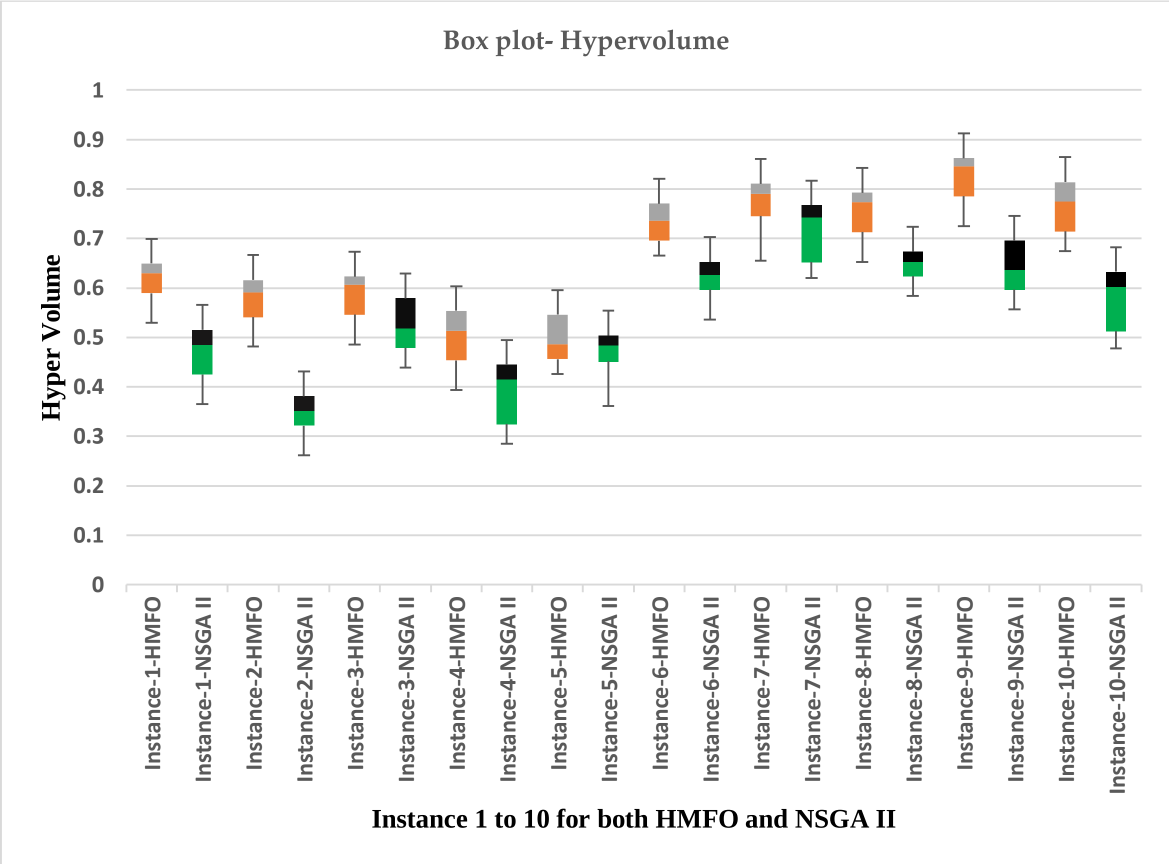


Figure. 4.28 Comparison of HMFO and NSGA-II with Hypervolume (HV) results for all the 10 instances of problems.

Apart from HV indicator, the results of various other performance measures for first five instances (1 to 5) and Last Five instances (6 to 10) are shown in Table 4.7 and Table 4.8 respectively. Out of all the available solutions the number of Non-Dominated (ND) solutions obtained by a proposed algorithm is denoted as α, the number of ND solutions that are not identified by the benchmark algorithm is denoted by β. For better results, it is suggested to have larger values of α and β and the ratio of β/α ratio near to one determines the effectiveness of the algorithm's strength. The percentage of ND solutions provided by a certain algorithm is expressed in terms of dominance ratio ϒ in Equation (4.14). If larger the dominance ratio indicates the superiority of the given algorithm.

$$\Upsilon = \frac{\left| B(\cup_{i} M_i) \setminus B(\cup_{i \neq k} M_i) \right|}{B(\cup_{i} M_i)} \tag{4.14}$$

where $\left|B(\bigcup_i M_i)\setminus B(\bigcup_{i\neq k} M_i)\right|$ indicates the non-dominated solutions found by the algorithm M that are unable to identify by the other standard algorithms.

K(a,b) in Equation (4.15) is useful for comparison of Pareto fronts particularly to identify the weak solutions of algorithm b with an algorithm (a>b) ultimately gives the correctness of the particular algorithm. Furthermore, lesser *K* species that the ND solutions identified by a particular algorithm are considered weaker.

$$K(a,b) = \frac{\left|b \in B, \exists a \in A : a > b\right|}{\left|B\right|} \tag{4.15}$$

Lesser π is necessary and the values which are very nearer indicates the highly distributed uniformly over the Pareto front. Equation (4.16) values of π which is the Euclidean length between end points of the identified ND Pareto set by an algorithm compared to the net ND Pareto front.

$$\Pi = \frac{F_f + F_l + \sum_{i=1}^{J-1}\left|F_i - \bar{F}\right|}{F_f + F_l + (J-1)\bar{F}} \tag{4.16}$$

λ indicates convergence power if smaller λ values are useful for identified ND solutions by the algorithm are falling very close range in the vicinity of net ND solutions for Euclidean lengths.

Hypervolume ratio or hyperarea ratio (HR) is the ratio of HV ratio of the algorithms. $HR_{(R,h,o,)}$= HV(R, o)/HV(h, o) from this lower HR is required to get better approximation [106]. From Table 4.7 and Table 4.8 it can be confirmed that the values marked in bold are the best values for showing the superiority of the proposed HMFO algorithm over the NSGA II algorithm.

Table 4.7 Results of performance indicators for comparison of HMFEO and NSGA II Instance 1 to 5.

Indicator	Algorithm	Instance				
		1	2	3	4	5
α	HMFEO	**8.5**	**10.8**	**9.7**	**10.6**	**8.8**
	NSGA II	8.0	10.5	9.4	10.6	8.6
β	HMFEO	**8.1**	**10.7**	**9.5**	**10.6**	**8.6**
	NSGA II	7.0	9.5	9.3	9.3	8.3
β/α	HMFEO	**0.9529**	**0.9900**	**0.9700**	**1.0000**	**0.9700**
	NSGA II	0.8750	0.9047	0.9893	0.8773	0.9651
ϒ	HMFEO	**0.5264**	**0.5079**	**0.6012**	**0.5000**	**0.5264**
	NSGA II	0.4736	0.4921	0.3988	0.5000	0.4736
K	HMFEO	**0.0800**	**0.0100**	**0.0300**	**0.0000**	**0.0300**
	NSGA II	0.1250	0.0953	0.0107	0.1227	0.0349
λ	HMFEO	**0.3693**	**0.0037**	**0.6289**	**0.0000**	**0.0042**
	NSGA II	12.123	9.236	8.265	8.6321	9.5632
π	HMFEO	0.4236	**0.4856**	**0.4982**	**0.4932**	**0.4495**
	NSGA II	**0.4125**	0.5563	0.6029	0.6765	0.5988
HR	HMFEO	**0.8526**	**0.7445**	0.9538	**0.6984**	**0.4495**
	NSGA II	1.2536	1.1456	**0.9469**	0.9548	0.9854

Table 4.8 Results of performance indicators for comparison of HMFEO and NSGA II Instance 6 to 10.

Indicator	Algorithm	Instance				
		6	7	8	9	10
α	HMFEO	**12.5**	**13.8**	**9.7**	**16.6**	**9.8**
	NSGA II	12.0	10.5	9.4	16.6	8.8
β	HMFEO	**12.1**	**12.7**	**9.5**	**16.6**	**9.6**
	NSGA II	11.0	9.2	9.3	15.0	8.3
β/α	HMFEO	**0.9682**	**0.9202**	**0.9700**	**1.0000**	**0.9795**
	NSGA II	0.9166	0.8761	0.9893	0.9036	0.9431
Ω	HMFEO	**0.5151**	**0.5070**	**0.5078**	**0.5000**	**0.5057**
	NSGA II	0.4848	0.4929	0.4922	0.5000	0.4942
K	HMFEO	**0.0800**	**0.0104**	**0.0310**	**0.0000**	**0.0400**
	NSGA II	0.1250	0.0835	0.0307	0.1027	0.0349
λ	HMFEO	**0.4693**	**0.0089**	**0.5259**	**0.0000**	**0.0063**
	NSGA II	12.123	9.236	8.265	8.6321	10.5632
π	HMFEO	0.5136	0.4856	**0.4982**	**0.4932**	**0.4445**
	NSGA II	**0.4125**	**0.4563**	0.5129	0.5365	0.5488
HR	HMFEO	**0.7415**	**0.8556**	**0.8629**	**0.7875**	**0.7589**
	NSGA II	0.9438	1.2465	1.1803	1.9888	0.8765

4.6 Conclusions

Advancements in technology, such as information and communication technologies (ICT) have changed the traditional manufacturing systems practices. This is especially true for a distributed manufacturing system due to its ability to cater to the needs such as Big data, interoperability, timely delivery, etc. In this research, the authors have considered a case study of automotive industries which are small and medium scale in nature and are geographically distributed with the objectives as a selection of appropriate suppliers according to product type and enhancing the system functions such as makespan, energy consumption and increase service utilization rate, interoperability, and reliability. To execute the first objective: supplier discovery is implemented through text mining based on supervised machine learning models. The results of classification Decision Tree (J48), Naïve Bayes, Random Forest, and Support Vector Machines are validated through various performance measures mainly Precision, Recall, and F-Measures. Decision trees have been found to be best with a precision of 0.93 for the purpose. These selected potential suppliers with the help of first objective in chapter 3 and their related information have been transferred as input data to the current work in chapter 4 phase.

The flexibility and complexity of a distributed manufacturing environment create the need for investigating the multiple process plans and multiple performance measures. Hence, the research work also investigated alternative process plans to the objective functions makespan, energy consumption, service utilization and reliability of services. we develop a MINLP model, and by acknowledging the NP-hard nature of the above scenario, a multi-objective evolutionary algorithm was decided to be utilized for which the input of task-specific suppliers is the outcome of supervised algorithmic models. As a result, we have used a Bio-inspired Moth Flame optimization evolutionary algorithm and tuned the algorithm to fit our problem objectives.

The results demonstrate that the use of evolutionary HMFO reduces the number of machine when compared to NSGA-II proving the effectiveness of the methodology used in this research. It also provides similar results with respect to the survivability of jobs as compared to NSGA-II. 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 HMFO. Various performance indicators are used to compare the superiority the proposed HMFO over the benchmark NSGA II algorithm. Thus, the results obtained showcase the superiority of the approach mentioned in this research.

Chapter 5

Energy efficient Network Manufacturing System with optimal process planning and scheduling

5. 1 Introduction

At present, research on manufacturing approaches and their implementation is intensely influenced by fast expansion in the field of communication and information technologies. On the other hand, markets should sustain the benefit of enormous market competition because of increased intricacy and functionality of vastly demanded products. The overall economic contest has certain advantages such as reduced cycle-time in the manufacturing system, ultimate data information, the standard flow of knowledge, etc. Current conventional manufacturing instances have to be renovated so that globally growing customer's demand can be satisfied [3]. Here, a Network Manufacturing System (NMS) has been considered as a newly recognized concept to gain the advantage over conventional manufacturing system.

Since most of the resources are non-renewable, acceleration of globalization, and rapid development in developing countries have led to increased consumption in energy resources. The industrial manufacturing sectors are consuming almost half of the energy delivered by the world (EIA, 2013). Countries like China, India, and Brazil facing global demand for a large variety of goods due to their relatively higher population growth and development in overall living standards. To fulfil their needs, the resources required are quite scarce. Hence, the efficient and sustainable utilization of resources has to be adopted, especially in the manufacturing sector [107], [108]. Sustainable development has been defined by the United Nations (1987) as development to satisfy the needs of the present without affecting the ability of future generations to meet their own needs by taking into account economic, social, and environmental dimensions.

In this research, a multi-objective network-based manufacturing model of customers, enterprise users, and a cloud of enterprises are developed to optimize makespan, machine utilization, and energy consumption while disclosing a product.

The purpose of the model is to obtain optimal process plans and scheduling of all jobs. The represented model has the following steps:

1) Developing mathematical equations by incorporating all constraints and relations to represent the problem.
2) Acquiring data and case study for giving input to the mathematical model.
3) Designing the framework of the optimization algorithm.

Problem descriptions containing suitable assumptions, mathematical models, and constraints are developed in Section 5.1 and 5.2. NSGA-II framework is discussed in Section 5.3. Section 5.4 consists of a demonstrative example for three cases. The results obtained are discussed in Section 5.5. Section 5.6 includes the conclusion of the chapter and discussion on the scope for future opportunities.

5.2 Problem description

A problem of the distributed network-based manufacturing system is considered, which is having a set of n jobs $\{J_1, ..., J_n\}$ of orders accepted from various customers, and m available machines $\{R_1, ... , R_m\}$. A particular set of schemes or substitute process plans are associated with each job J_i and a series of sequential V_i operations ($O_{il}, ..., O_{iv_i}$) is linked to each process plan shown in Fig. 5.1. Consequently, the available machines are employed to process jobs with different possible process plans at different enterprises to achieve better use of resources and satisfactory delivery schedules. Each machine can be operated with different speeds due to their dispersion over geographically distributed enterprises, thus each task is associated with an integer energy E_{Oil} and duration P_{Oil} used by the corresponding machine. The association between duration and energy can be expressed as "Job(Speed) || Makespan, Energy" for this problem. Each task is operated with altered speeds and each speed results in a specific processing time and energy consumption. The operation time decreases with the increase in working speed which also results in increased energy consumption. In this chapter, different jobs, their predecessor and successor operations, machine candidates, processing time, energy efficiency have been considered. The objective is to determine the best suitable enterprise and feasible schedule which combines minimization of the makespan and the energy consumed by the machines with maximized resource utilization. Hence, the dead time can be retrieved by escalating the machine's speed if a task is delayed to recover the original solution. The mentioned network-based manufacturing case is one of the intricate problems of the current Instance where a vital role is played by the servicing operating time and transporting time between two corresponding machines to sustain process planning and scheduling functions. Being a large research space, the problem becomes intricate for resulting out the best suiting optimal solution. Therefore, IPPS provides the prospective to produce an effectual optimized process plan due to flexibility in networked manufacturing. Hence, the origination of optimal process plans related to every job linked with constraints becomes a demanding task so; it can be taken as a new problem on the report of current manufacturing circumstances. The Table 5.1 shows various notations in the considred mathematical model.

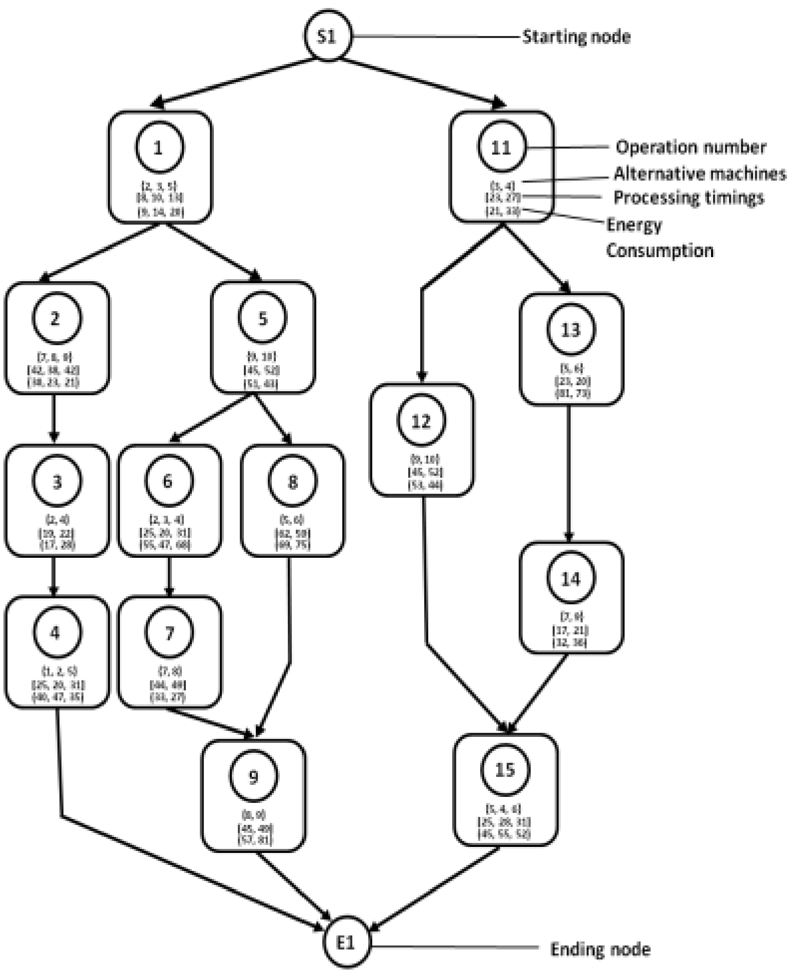


Figure 5.1 Sample Flow chart for flexible process plan.

5.2.1 Assumptions

1. Job pre-emption is not permitted.
2. Any task of a job being operated on any available machine should be completed without any interruption until it finishes.
3. Only one job is to be handled by each machine at a point time.
4. All jobs and machines can go under process on-time zero.
5. A linear relation between energy consumption and speed is considered.
6. A linear relation between energy consumption and speed is considered.

Table 5.1. Notations used in the mathematical model

Parameters	Description
N	total number of jobs
S	total number of machines
P_j	the total number of alternative process plans for j^{th} job
V_{jp}	the number of operation in the pth alternative process plan for jth job
O_{jpo}	the oth operation in the pth alternative process plan for jth job
T_j	makespan of jth job from the set of possible process plans
Pt_{jpom}	the processing time of the operation Ojpo on mth machine
θ	a very large positive number
C_j	the completion time of job j
C_{jpom}	the earliest completion time of operation O_{jop} on machine m
mpt_s	The total processing time of s^{th} machine
mct_s	The closing time of s^{th} machine
mst_s	The starting time of s^{th} machine
S	The number of machines
N	The number of jobs
V_{jp}	The total number of the operations in p^{th} process plan for j^{th} job
R_{jpo}	The available machines for operation O_{jpo}
W_m	The processing power for m^{th} machine
I_m	The idle power for m^{th} machine

$\mu_m(t)$	The input power function for m^{th} machine wrt. time t
O_{jpo}	The o^{th} operation in the p^{th} alternative process plan for j^{th} job
E_{jpom}	The ending time of operation O_{jpo} on m_{th} machine
Cmax	The makespan
$\emptyset$	The total energy consumption
$\emptyset_{jpo}$	The energy consumption for the operation O_{jpo}
$\emptyset_{pm}$	The total processing energy consumption of m^{th} machine
$\emptyset_{im}$	The idle state energy consumption of of m^{th} machine
$\emptyset_{0/1m}$	The turning-on/off energy consumption for m^{th} machine
D_{m1}	The time duration of turn on for m^{th} machine
D_{m2}	The time duration of turn off for m^{th} machine
F_m	The time at which machine m is to be turned on at the starting.
G_m	The time at which machine m is to be turned off at the finishing.
$\emptyset_{sm}$	The stand by energy consumption of of m^{th} machine
Z	The total turning-on/off numbers for all machines in pth process plan
z_m	The total turning-on/off numbers for m^{th} machine
F_{m1}	The latest time to turn on machine k
B_m	The breakeven duration to determine if m^{th} machines in the idle state can
L_m	A threshold time limit for m^{th} machine to be turned off
Y_{sm}	Stand by power for m^{th} machine

5.2.2. Decision Variables

$$\alpha_{jp} = \begin{cases} 1\text{ ; if the } p^{th} \text{flexible process plan of } j^{th} \text{ job is selected} \\ 0\text{ ; otherwise} \end{cases}$$

$$\beta_{jpoQrsm} = \begin{cases} 1\text{ ; the operation } U_{jpo} \text{ precedes the operation } O_{Qrs} \text{ on machine m} \\ 0\text{ ; otherwise} \end{cases}$$

$$\gamma_{jpom} = \begin{cases} 1\text{ ; if machine } m \text{ is selected for } O_{jpo} \\ 0\text{ ; otherwise} \end{cases}$$

$$A_{jpom} = \begin{cases} 1 \text{ , if the operation } O_{jpo} \text{ is being processed on } m^{th} \text{ machine} \\ 0 \text{ , otherwise} \end{cases}$$

$$G_{jpoqrkm} = \begin{cases} 1 & \text{if operation } O_{jpo} \text{ is the successor of operation } O_{qrk} \text{ on } m^{th} \text{ machine} \\ 0 & \text{if operation } O_{jpo} \text{ and operation } O_{qrk} \text{ are not adjacent} \\ -1 & \text{if operation } O_{jpo} \text{ is preceding operation } O_{qrk} \end{cases}$$

$$D_{jpom} = \begin{cases} 1 \text{ , if } m^{th} \text{ machine is to be turned off between the operation } O_{jpo} \\ \text{ and operation } O_{qrk} \\ 0 \text{ , if } m^{th} \text{ machine to be turned on between the operation } O_{jpo} \\ \text{and operation } O_{qrk} \end{cases}$$

5.3 Mathematical Modelling of the considered Networked Manuacturing System.

The desired outcomes of this problem are minimization of makespan, maximization of machine utilization, and minimization of energy consumption as represented as follows:

Objectives:

Makespan minimisation: T_j=Max C_{jpom} (5.1)

Maximization of machine utilization: $mu = \frac{\sum_{s=1}^{m} mpt_s}{\sum_{s=1}^{m} (mct_s - mst_s)}$ (5.2)

Minimization of energy consumption: $\varnothing = \sum_{m=1}^{S} (\varnothing_{0/1m} + \varnothing_{im} + \varnothing_{pm} + \varnothing_{sm})$ (5.3)

The energy consumption model

It can be divided into four operating states as follows.

a. ***The turning on/off state***

$$\text{Ø}_{0/1m} = z_m \left[\int_{F_m}^{F_m+D_{m1}} \mu_m(t)\,dt + \int_{G_m-D_{m2}}^{G_m} \mu_m(t)\,dt \right] \max_{j,o}(X_{ijkq}) \tag{5.4}$$

$$F_m = \min_{j,o}(A_{jpom}(E_{jpom} - Pt_{jpom})) \tag{5.5}$$

$$G_m = \max_{j,o}(A_{jpom}E_{jpom}) \tag{5.6}$$

b. ***The idle state***

$$\text{Ø}_{im} = \sum_{j,q}^{N} \sum_{o,k}^{\max(V_{jp},V_{qr})} \text{Ø}_{im\,jpoqrk} \tag{5.7}$$

$$\text{Ø}_{imjpoqrk} = \begin{cases} Q(Q_1(\max(F_{m1}+L_m, E_{jpom}) - E_{jpom}) + \\ (Q_2(\max(F_{m1}+L_m, E_{qrkm}) - E_{qrkm}), D_{jpoqrk}=1 \\ Q(Q_1(E_{qrkm} - Pt_{qrkm} - E_{jpom}) + Q_2(E_{jpom} - Pt_{jpom} - E_{qrkm})),\ D_{jpoqrk}=0 \end{cases} \tag{5.8}$$

Where $Q = A_{jpom}A_{qrkm}(G_{jpoqrkm}/2)$, $Q_1 = I_m(G_{jpoqrkm} - 1)$ and $Q_2 = I_m(G_{jpoqrkm} + 1)$

$$F_{m1} = \begin{cases} \min_{a,b}(E_{aebm} - Pt_{aebm})A_{aebm}A_{jpom}A_{qrkm},\ D_{aebqrk} + U_{aebjpo} = 1 \cap U_{jpoqrk} = 1 \cap 0 < E_{aebm} < K \\ F_m,\ otherwise \end{cases} \tag{5.9}$$

$$K = (G_{jpoqrkm}/2)(G_{jpoqrkm}+1)(E_{qrkm} - Pt_{qrkm}) + (G_{jpoqrkm}-1)(E_{jpom} - Pt_{jpom}) \tag{5.10}$$

c. ***The processing state***

$$Ø_{pm} = W_m (\sum_{j}^{N} \sum_{o}^{V_{jp}} A_{jpom} Pt_{jpom}) \tag{5.11}$$

d. The standby state

$$Ø_{sm} = C_{\max} Y_{sm} \tag{5.12}$$

Constraints:

For the initial operation in the p^{th} process plan for j^{th} job:

$$C_{jp1m} + \theta(1-\alpha_{jp}) \geq Pt_{jp1m} \tag{5.13}$$

For the final operation in the p^{th} process plan for j^{th} job:

$$C_{jv_{jp}jp} - \theta(1-\alpha_{jp}) \leq C_{jpom} \tag{5.14}$$

A job's different operations cannot be processed simultaneously

$$C_{jpom} - C_{jp(o-1)m} + \theta(1-\alpha_{jp}) \geq Pt_{jpom} \tag{5.15}$$

Only one job is to be handled by each machine at a point of time

$$C_{jpom} - C_{Qrsm} + \theta\beta_{jpoQrsm} \geq Pt_{jpom} \tag{5.16}$$

Each job can be associated with single process plan

$$\sum_{p=1}^{P_j} \alpha_{jp} = 1 \tag{5.17}$$

Each operation can be process on a single machine only

$$\sum_{m=1}^{S} \gamma_{jpom} = 1 \tag{5.18}$$

$$E_{jpo} - Pt_{jpo} \geq E_{jp(o-1)} \tag{5.19}$$

$$\begin{aligned} &(G_{jpoqrkm}/2)(G_{jpoqrkm}-1)(E_{qrk}-E_{jpo}-Pt_{qrkm})A_{jpom}A_{qrkm} \\ &+(G_{jpoqrkm}/2)(G_{jpoqrkm}+1)(E_{jpo}-E_{qrk}-Pt_{jpom})A_{jpom}A_{qrkm} \geq 0 \end{aligned} \tag{5.20}$$

$$\sum_{m=1}^{S} A_{jpom} = 1 \tag{5.21}$$

$$E_{jpom} \geq 0 \tag{5.22}$$

$$Pt_{jpom} \geq 0 \tag{5.23}$$

This problem's objectives are to primarily emphasize on job scheduling so that maximum of their total completion time of all operations can be minimized, i.e., makespan as specified in equation (5.1); maximization of the machine or resource utilization as referred in equation (5.2); and to minimize the Energy consumption of machines as specified in equation (5.3). The problem is subjected to certain constraints which are listed in the equations (5.13)-(5.23). Constraint (5.13) expresses restriction in processing various operations related to a job concurrently for alternative process plans. Constraint (5.14) states that only a single operation can be operated on any machine at a time. Constraint (5.15) comprises that each job can be related to only one single process plan. Constraint (5.16) shows that a single machine is to be selected for each operation. The precedence relation is depicted in Eq. (5.19). Constraint (5.20) states that only one job can be processed by any machine at a point of time. Constraint (5.21) represents a decision variable that is used to assign availability of the machines at a particular time. Practicality of our problem is portrayed in (5.22) and (5.23).

5.4 Non Dominataed Sorting Genetic Algorithm NSGA- II

The theory of the NSGA concept was first proposed by [109] and [110] implemented it in their work. Working with the selection operator is the main difference between NSGA and simple genetic algorithms. Various difficult multi-objective problems have been solved so far by exploiting the concept of NSGA shown in Fig**.** 5.2 but still, it has certain limitations as complexity in the computation of non-dominated sorting, absence of elitism, and the necessity of having a particular tune-able parameter. The above-indicated shortcomings have been removed by introducing NSGA-II up to a certain extent. The mutation rate and crossover rate are the probabilities on which the parents are offspring mutated and crossbred, respectively. The values of the required parameters have been shown in Table 5.2.

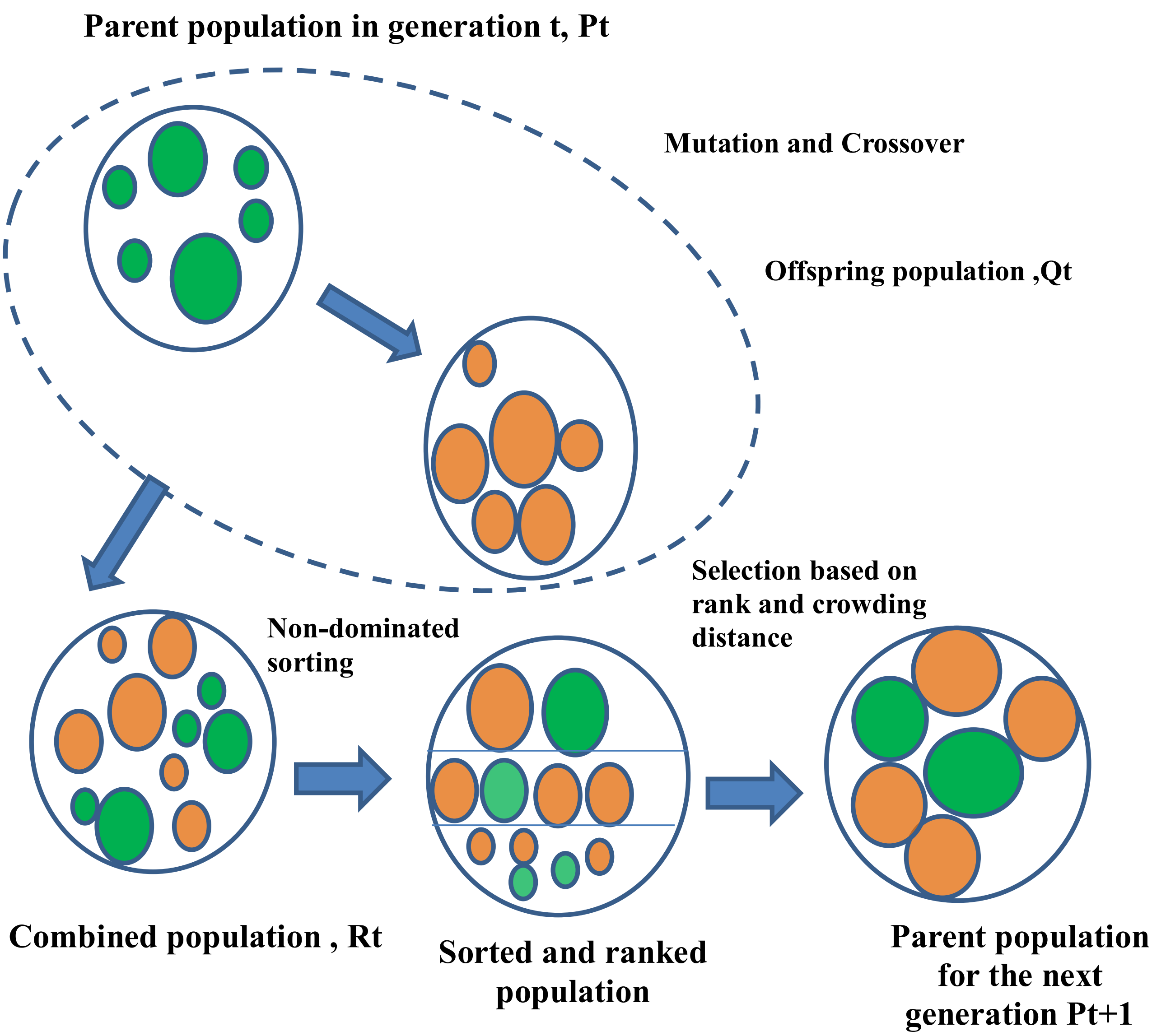


Figure 5.2. Working of Non dominated Sorting Genetic Algorithm (NSGA II)

5.5 Controlled elitist NSGA-II (CE-NSGA-II)

Quick erasure of non-elitist front solutions and lack of variety in some decision variables are undesired outcomes of NSGA-II, which leads to the significance of the much superior CE-NSGA-II algorithm. The following sections describe various stages of the CE-NSGA-II algorithm in the reference of networked manufacturing problems having multi-objectives and above-disclosed outcomes of NSGA-II. The framework for CE-NSGA II shown in the Fig. 5.3.

Table 5.2 Parameter values for algorithm.

Parameter	Parameter Value
Population Size	50
Maximum Generations	50
Mutation Probability (M_p)	0.05
Cross-over Probability (C_p)	0.84

5.5.1 Population Initialization

At first, the initial population is generated arbitrarily for specified population size. The process also plays a major role in obtaining more optimal solutions. It has been proven that the initial populations generated may affect the best value of the objective function and that their effects may last for several generations [48].

5.5.2 Evolutionary Operations

In the process of evolution, genetic variation is necessary. The operations that are performed are analogous to those happening in nature in the process of evolution: Survival of the fittest or selection, mutation, and crossover (also called as reproduction or recombination). These operations are carried out to protect the diversity in the population *N* and to create a new child population of the same size.

5.5.3 Selection

In this operation, better solutions are given more preference which allows them to pass on their genes to the next generation during the execution of the algorithm. This is the preliminary step before performing cross-over or mutation. The best solutions are selected using the fitness values of the objective functions. The fitness value represents the closeness of the solution in achieving the specified objective. There are various methods of selection used for different applications. In our case, we have used a tournament selection in the algorithm.

5.5.4 Mutation

The mutation is used to develop the solution space by generating neighbors in different directions. It can be executed in four methods as swap, insertion, displacement, and inversion where the first two produce close neighbors while the other two create distant neighbors. This operation protects the robust intermediate solutions by diversification and to adjust the fragile ones. Here bit wise mutation is employed with P_m (mutation probability) as 0.05 Table 5.2.

5.6 Validation of proposed nodel with the help of cplex for small sized instances.

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 seconds was taken into account for solving the test instances [112] mentioned in Table 5.3. A comparison of cplex results with the considered CE NSGA II was shown in Table 5.3. All the three objectives namely makespan, energy consumption, machine utilization values obtained by an augmented e- constraint method followed in CPLEX solver of GAMS [113] and the results are compared with the objective values of proposed CE NSGA II. The lower makespan and energy consumption and higher energy consumption values of proposed CE NSGA II indicates the better performance of the algorithm over all the test Instances shown in Table 5.3. Moreover, large size problem Instances are not able to solve by using exact solution methodologies like CPLEX, taking huge amount of CPU times.

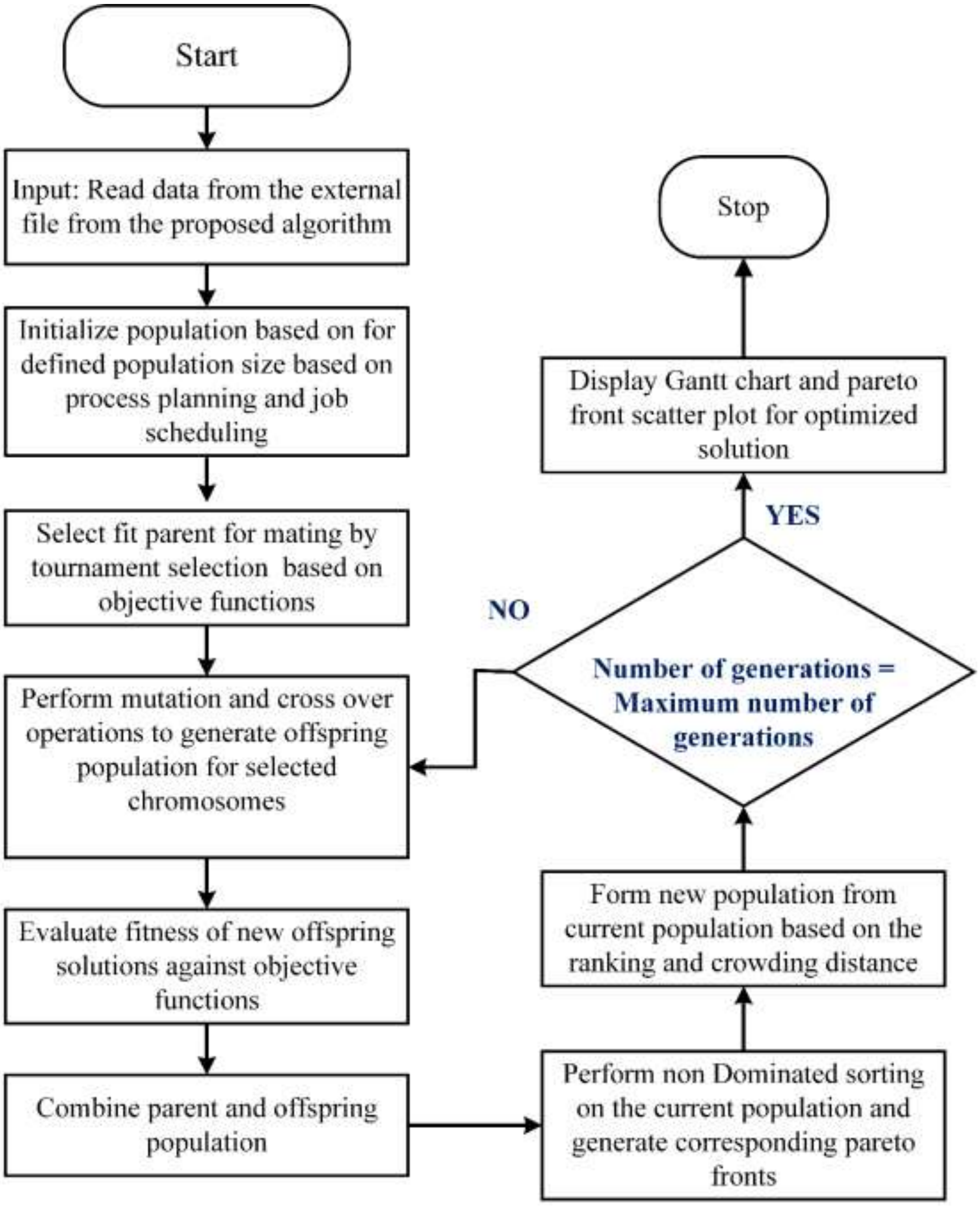


Figure 5.3 CE-NSGA-II Framework.

Table 5.3 Validation of proposed model with the help of cplex for all the test instances 1 to 10.

Test Instance	(n*m)	MILP by cplex solver				CE-NSGA II			
		Makepsan (Minutes)	EC(Kw)	MU (%)	CPU Time (Sec)	Makepsan (Minutes)	EC(Kw)	MU (%)	CPU Time (Sec)
Test Instance 1	2*2	22	426	0.6	210	21	402	0.62	20
Test Instance 2	2*2	19	329	0.54	220	16	312	0.61	32
Test Instance 3	3*2	26	521	0.51	292	22	496	0.56	28
Test Instance 4	3*2	35	689	0.53	284	22	641	0.56	37
Test Instance 5	3*2	39	773	0.59	292	31	662	0.67	36
Test Instance 6	3*2	34	664	0.53	361	29	623	0.64	36
Test Instance 7	3*3	39	756	0.54	792	32	699	0.62	37
Test Instance 8	3*4	44	863	0.56	1260	36	756	0.59	34
Test Instance 9	3*5	41	689	0.58	1505	29	593	0.63	30
Test Instance 10	4*5	43	723	0.51	2163	32	652	0.61	39

5.7 Demonstrative example of Energy efficient NMS

Three illustrative instances (represented by n x m / Job X Machine problem) have been chosen as a testbed to represent the performance and efficacy of the proposed algorithm. Available machine and processing time data are collected from the industry, and the energy consumption is computed through the mathematical model using practical machine power and speed parameters. CE NSGA II algorithm is applied to obtain optimized process planning and scheduling with minimum makespan, maximum machine utilization, and minimum energy consumption. Table 5.2 consists of all the necessary parameters for the proposed algorithm.

Tables 5.4, 5.5**,** have the input data to the algorithm for the 6 *6 and 6*8 cases respectively. Three different brackets are used to represent the parameter's data. Available machines, processing time, and energy consumption are placed in curly, square, and round brackets respectively. For example, j_3 job's O_3 has available machines as {2, 5}, corresponding [5, 6] minutes processing time, and (372, 378) kilo-joule energy consumption.

5.7.1 Case 1 (6 X 6 problem)

This Instance exhibits the order placed by costumers for six different jobs where the corresponding operations are to be processed by six geographically located machines at different enterprises. Anyone of multiple process plans can be employed for each job and multiple machines are available to process each operation. Popularly applied scheduling representation means i.e., Gantt chart is characterized in Fig. 5.4, which pictures the allotment of operations on various available machines, beginning and ending time of each operation, and the optimized process plan

Time and machines are represented on *X* and *Y*-axis of the Gantt chart. The output results for makespan selected optimal process plan and job scheduling obtained from the algorithm for 6 X 6 case are shown in Table 5.4.

5.7.2 Case 2 (8 X 8 problem)

Case of eight different jobs from clients to be processed at eight machines available in the network manufacturing system is explored in Instance 2. The output results for makespan selected optimal process plan and job scheduling obtained from the algorithm for 8 X 8 case are shown in Table 5.5. The job scheduling Gantt chart is presented in Fig. 5.5.

5.7.3 Case 3 (6 X 8 problem)

This Instance consists of six different jobs to be completed through processing operations using eight available machines. The output results for makespan selected optimal process plan and job scheduling obtained from the algorithm for 6 X 8 are shown the Gantt chart is shown in Fig. 5.6.

Table 5.4 Input data for 6 jobs 6 machines problem considered in NMS.

Input data for 6X6 problem	
	Operations

Job	PP	O_1	O_2	O_3	O_4	O_5	O_6
J_1	$PP_{1,1}$	{1, 2}	{3, 4, 5}	{6}			
		[6, 5]	[7, 6, 6]	[8]			
		(493, 372)	(496, 410,	(619)			
	$PP_{1,2}$	{1, 3}	{2, 4}	{3, 5}	{4, 5, 6}		
		[4, 5]	[4, 5]	[5, 6]	[5, 5, 4]		
		(329, 354)	(298, 342)	(354, 378)	(342, 315,		
J_2	$PP_{2,1}$	{2}	{1, 3}	{2, 4, 6}	{3, 5}	{2, 4}	{4, 6}
		[4]	[2, 3]	[4, 3, 5]	[2, 4]	[3, 4]	[3, 5]
		(298)	(164, 212)	(298, 205,	(142, 252)	(223, 274)	(205, 387)
	$PP_{2,2}$	{1, 3, 5}	{4}	{4, 6}	{4}	{4, 6}	{1, 6}
		[1, 5, 7]	[5]	[1, 6]	[4]	[1, 2]	[5, 6]
		(82, 354, 441)	(342)	(68, 464)	(274)	(68, 155)	(411, 464)
J_3	$PP_{3,1}$	{2, 3}	{1, 4}	{2, 5}	{3, 6}	{1, 6}	{5}
		[5, 6]	[6, 5]	[5, 6]	[6, 5]	[6, 6]	[4]
		(372, 425)	(493, 342)	(372, 378)	(425, 387)	(493, 464)	(252)
	$PP_{3,2}$	{1}	{3, 4}	{5}			
		[7]	[8, 8]	[9]			
		(575)	(566, 547)	(567)			
	$PP_{3,3}$	{2, 3}	{4}	{3, 5}	{4, 6}	{1, 2}	
		[7, 6]	[7]	[7, 8]	[7, 8]	[1, 4]	
		(521, 425)	(479)	(496, 504)	(479, 619)	(82, 298)	
J_4	$PP_{4,1}$	{1, 2}	{3, 4}	{6}	{1}		
		[7, 8]	[7, 6]	[9]	[5]		

		(575, 595)	(496, 410)	(697)	(411)		
	$PP_{4,2}$	{1, 3, 5}	{2}	{3, 4, 6}	{5, 6}		
		[4, 3, 7]	[4]	[4, 5, 6]	[3, 5]		
		(329, 212, 441)	(298)	(283, 342,	(189, 387)		
J_5	$PP_{5,1}$	{1}	{2, 4}	{3}	{5, 6}		
		[3]	[4, 4]	[4]	[3, 3]		
		(247)	(298, 274)	(283)	(189, 232)		
	$PP_{5,2}$	{2, 4}	{5}	{3, 6}			
		[5, 6]	[7]	[9, 8]			
		(372, 410)	(441)	(437, 619)			
J_6	$PP_{6,1}$	{1, 2}	{3, 4}	{2, 5}	{3}	{4, 5}	{3, 6}
		[3, 4]	[4, 3]	[5, 3]	[4]	[5, 6]	[5, 4]
		(247, 298)	(283, 205)	(372, 189)	(283)	(343, 378)	(354, 310)
	$PP_{6,2}$	{1, 3}	{2, 3}	{2, 4}	{6}		
		[4, 4]	[5, 6]	[6, 7]	[7]		
		(329, 283)	(372, 425)	(446, 479)	(542)		
	$PP_{6,3}$	{1, 2, 3}	{4, 5}	{3, 6}			
		[3, 5, 8]	[7, 10]	[9, 9]			
		(247, 372, 566)	(479, 630)	(637, 697)			

Table 5.5 Input data for 6 jobs 6 machines problem considered in NMS.

Input data for 6X8 problem						
		Operations				
Job	**PP**	**O_1**	**O_2**	**O_3**	**O_4**	**O_5**
J_1	$PP_{1,1}$	{2, 4}	{7, 8}	{1, 2}	{8, 6}	{3, 8}

		[18, 22]	[39, 36]	[11, 10]	[31, 34]	[26, 24]
		(1339, 1505)	(3346, 2138)	(904, 744)	(1841, 2632)	(1841, 1426)
	$PP_{1,2}$	{2, 4}	{3, 5}	{1, 2, 4}	{5, 6}	{3, 8}
		[18, 22]	[21, 23]	[10, 12, 15]	[32, 30]	[26, 24]
		(1339, 1505)	(1487, 1449)	(822, 893, 1026)	(2016, 2322)	(1841, 1426)
	$PP_{1,3}$	{2, 4}	{3, 5}	{1, 7}	{3, 8}	
		[18, 22]	[21, 23]	[45, 44]	[26, 24]	
		(1339, 1505)	(1487, 1449)	(3699, 3775)	(1841, 1426)	
J_2	$PP_{2,1}$	{2, 4}	{7, 8}	{1, 2}	{3, 8}	
		[18, 22]	[39, 36]	[37, 39]	[26, 24]	
		(1339, 1505)	(3346, 2138)	(3041, 2902)	(1842, 1426)	
	$PP_{2,2}$	{2, 4}	{8, 6}	{1, 2, 4}	{5, 6}	{3, 8}
		[18, 22]	[20, 21]	[10, 12, 15]	[36, 38]	[26, 24]
		(1339, 1505)	(1188, 1625)	(822, 893, 1026)	(2268, 2941)	(1841, 1426)
	$PP_{2,3}$	{2, 4}	{8, 6}	{1, 7}	{3, 8}	
		[18, 22]	[20, 21]	[45, 44]	[26, 24]	
		(1339, 1505)	(1188, 1625)	(3699, 3775)	(1842, 1426)	
	$PP_{2,4}$	{2, 4}	{3, 5}	{1, 2, 4}	{5, 6}	{3, 8}
		[18, 22]	[21, 23]	[10, 12, 15]	[36, 38]	[26, 24]
		(1339, 1505)	(1487, 1449)	(822, 893, 1026)	(2268, 2941)	(1841, 1426)
	$PP_{2,5}$	{2, 4}	{3, 5}	{1, 7}	{3, 8}	
		[18, 22]	[21, 23]	[45, 44]	[26, 24]	
		(1339, 1505)	(1487, 1449)	(3699, 3775)	(1842, 1426)	
J_3	$PP_{3,1}$	{1, 4}	{6, 7}	{5, 8}	{2, 4}	
		[22, 25]	[24, 22]	[20, 19]	[22, 27]	
		(1808, 1710)	(1858, 1888)	(1260, 1129)	(1637, 1847)	
	$PP_{3,2}$	{3, 5}	{4, 6}	{2, 3, 5}	{2, 4}	
		[12, 15]	[24, 23]	[30, 31, 24]	[22, 27]	

		(850, 945)	(1642, 1780)	(2232, 2195, 1512)	(1637, 1847)	
	$PP_{3,3}$	{3, 5}	{6, 7}	{1, 8}	{2, 4}	
		[12, 15]	[21, 22]	[32, 30]	[22, 27]	
		(850, 945)	(1625, 1888)	(2630, 1782)	(1637, 1847)	
J_4	$PP_{4,1}$	{1, 4}	{6, 7}	{2, 4}		
		[22, 25]	[42, 44]	[22, 27]		
		(1808, 1710)	(3251, 3775)	(1637, 1847)		
	$PP_{4,2}$	{1, 4}	{5, 8}	{2, 4}		
		[22, 25]	[41, 43]	[22, 27]		
		(1808, 1710)	(2583, 2554)	(1637, 1847)		
	$PP_{4,3}$	{3, 5}	{4, 5}	{2, 3, 5}	{2, 4}	
		[12, 15]	[24, 23]	[30, 31, 29]	[22, 27]	
		(850, 945)	(1642, 1449)	(2232, 2195, 1827)	(1637, 1847)	
	$PP_{4,4}$	{3, 5}	{6, 7}	{1, 8}	{2, 4}	
		[12, 15]	[21, 22]	[32, 30]	[22, 27]	
		(850, 945)	(1625, 1888)	(2630, 1782)	(1637, 1847)	
J_5	$PP_{5,1}$	{2, 4}	{1, 3}	{6, 7}	{5, 8}	{3, 4}
		[18, 22]	[22, 25]	[24, 22]	[20, 18]	[22, 27]
		(1339, 1505)	(1808, 1770)	(1858, 1888)	(1260, 1069)	(1558, 1847)
	$PP_{5,2}$	{2, 4}	{3, 5}	{6, 8}	{1, 7}	{3, 4}
		[18, 22]	[12, 15]	[19, 21]	[32, 31]	[22, 27]
		(1339, 1505)	(850, 945)	(1471, 1247)	(2630, 2660)	(1558, 1847)
	$PP_{5,3}$	{2, 4}	{3, 5}	{1, 2, 6}	{3, 4}	
		[18, 22]	[12, 15]	[50, 52, 54]	[22, 27]	
		(1339, 1505)	(850, 945)	(4110, 3869 , 4180)	(1558, 1847)	
J_6	$PP_{6,1}$	{2, 4}	{1, 7}	{6, 7}	{5, 8}	{3, 4}
		[18, 22]	[22, 24]	[24, 22]	[20, 18]	[22, 27]

		(1339, 1505)	(1808, 2059)	(1858, 1888)	(1260, 1069)	(1558, 1847)
	$PP_{6,2}$	{2, 4}	{1, 3}	{6, 7}	{5, 8}	{3, 4}
		[18, 22]	[21, 25]	[24, 22]	[20, 18]	[22, 27]
		(1339, 1505)	(1726, 1770)	(1858, 1888)	(1260, 1069)	(1558, 1847)
	$PP_{6,3}$	{2, 4}	{3, 5}	{6, 8}	{3, 4}	
		[18, 22]	[12, 15]	[53, 51]	[22, 27]	
		(1339, 1505)	(850, 945)	(4102, 3029)	(1558, 1847)	
	$PP_{6,4}$	{2, 4}	{3, 5}	{1, 2, 6}	{3, 4}	
		[18, 22]	[12, 15]	[50, 52, 54]	[22, 27]	
		(1339, 1505)	(850, 945)	(4110, 3869 , 4180)	(1558, 1847)	

5.8 Results and discussion

The parameters deciding the performance of the projected algorithm are the utilization of machines, makespan, number of generations, and computational run time. Three incompatible objectives are selected in our research work as minimization of makespan, minimization of energy consumption, and maximization of machine utilization. Better efficiency of the algorithm and higher productivity of the manufacturing system can be assured with lesser makespan value.

Table 5.6 Job scheduling for 6X6 case

	Case 3 (6 X 8)		
Job No	**Makespan (Min)**	**Process plan**	**Scheduling of jobs**
1	107	3	[2,3,7,8]
2	109	5	[2,3,7,3]
3	105	2	[5,6,2,4]
4	94	3	[3,4,3,4]
5	116	3	[4,3,6,4]
6	105	4	[2,5,1,3]

Table 5.7 Job scheduling for 6X8 case

	Case 1 (6 X 6)		
Job No	**Makespan (Min)**	**Process plan**	**Scheduling of jobs**
1	20	2	[3,2,5,5]
2	18	2	[1,4,4,4,4,6]
3	24	2	[1,4,5]
4	22	2	[5,2,6,6]
5	20	2	[2,5,6]
6	24	2	[3,3,4,6]

In Tables 5.6, 5.7, and, 5.8 the total makespan i.e., the total of processing times on each machine for each job; the optimal process plan that is selected from the set of alternatives provided and the scheduling of jobs as to "on which machine which operation is performed" has been provided for all the tree cases. The makespan, energy consumption, machine utilization, and computational time for fittest solutions are detailed in Table 5.9. It can be referred that the proposed algorithm converges fast because it encounters termination norms in a lower number of generations. Though, obtaining lower computational time with fewer generations is not guaranteed because each algorithm has particular intricacy implicated for every single run of programming.

This occurrence leads to the concern of computational time and the proposed algorithm deals with several generations and computational time in an equivalent manner and provides justified computational time so that reaction of manufacturing resources is improved through quick process plan generation. Energy efficiency and cost-effectiveness of any manufacturing sector can be improved with lower energy consumption through proper process plan and scheduling selection which can be inferred from our results. One more crucial factor is the superior consumption of manufacturing resources which is termed as "Machine Utilization" and can be defined as the percentage ratio of real operation processing time to the overall running time of all the machines.

Table 5.8 Job scheduling for 8X8 case

	Case 2 (8 X 8)		
Job No	**Makespan (Min)**	**Process plan**	**Scheduling of jobs**
1	22	2	[1,2,3,8]
2	24	2	[3,4,4,4,4,8,4]
3	23	2	[1,3,8]
4	19	2	[1,2,6,6]
5	14	1	[1,2,3,8]
6	22	2	[3,3,7,6]
7	23	1	[2,4,6,5,6]
8	18	2	[2,4,6]

Table 5.9 Overall optimized value of objective functions

Experiment	Makespan (Min)	Energy Consumption (KJ)	Machine Utilization (%)	Execution Time (Sec)
Case 1 (6X6)	43	9083	94.82	4.2377
Case 2 (8X8)	47	11889	90.16	5.6243
Case 3 (6X8)	173	46142	91.12	5.1098

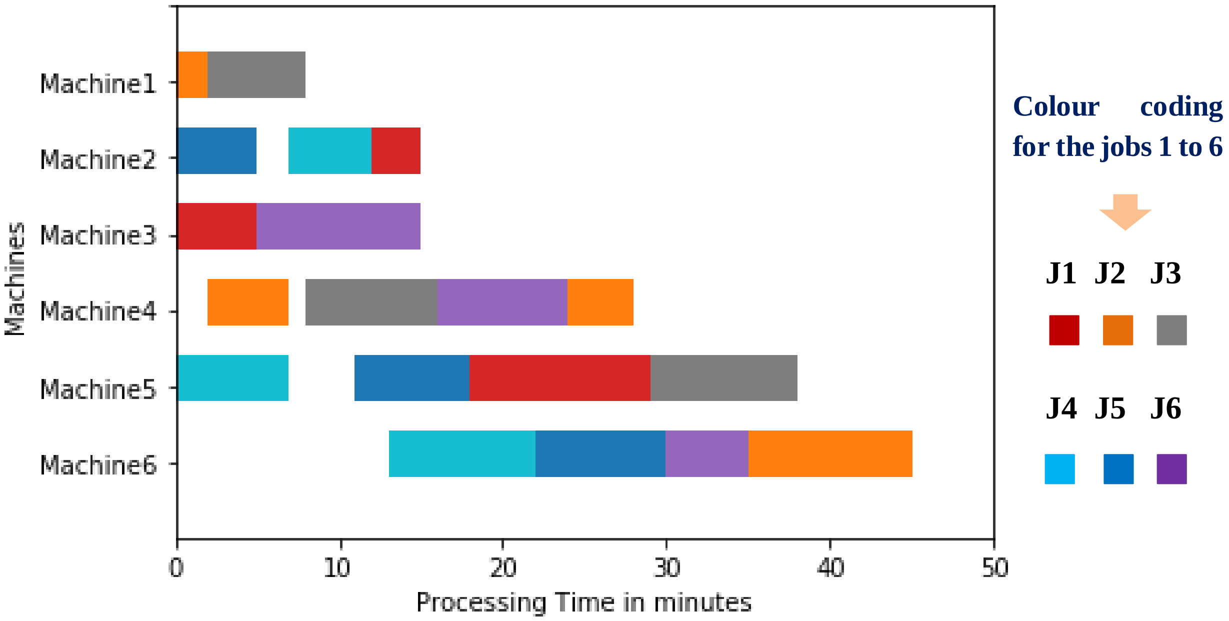


Figure 5.4 Gantt chart showing job scheduling for 6X6 case

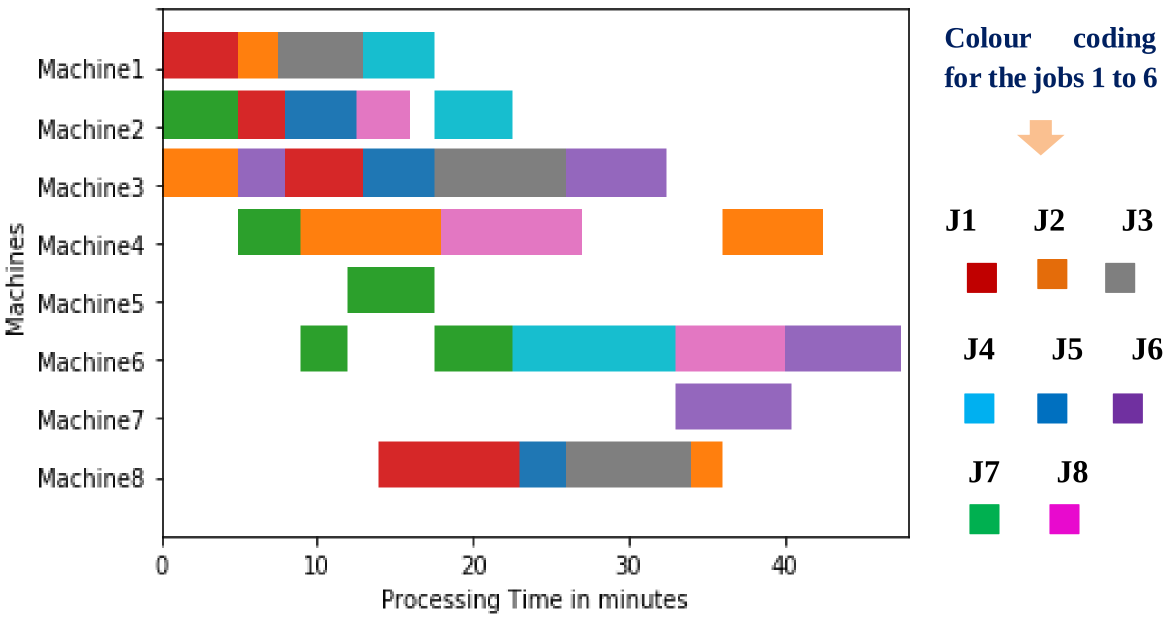


Figure 5.5 Gantt chart showing job scheduling for 8X8 case

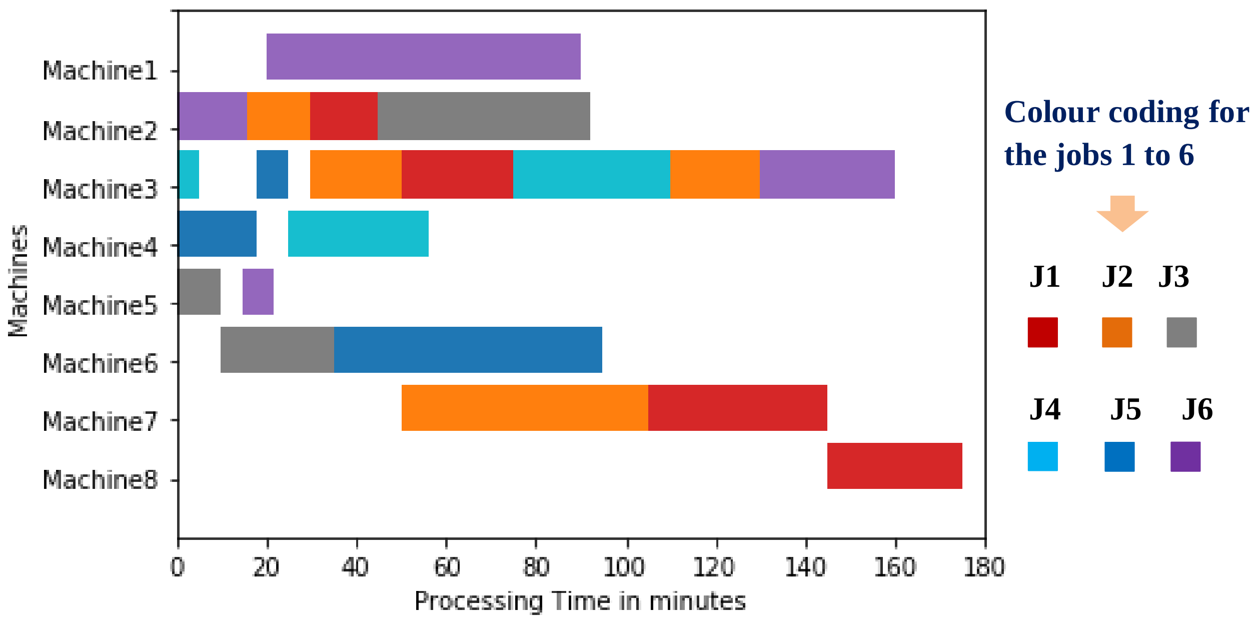


Figure 5.6 Gantt chart showing job scheduling for 6X8 case

The scatter plots of the optimal Pareto-fronts for the three cases have been shown in Fig. 5.7, 5.8, and 5.9. The dark highlighted points show the locus of the optimal Pareto-fronts obtained. A Pareto-front contains all reasonable solutions based on the objective functions and constraints. Although there are various methods to choose the best solution; one of the simplest and easiest ways is to compare with the ideal point and recognize the point closest to the ideal point. Also, the solution chosen from the frontiers depends on the perspective of the decision-maker.

The black colour solution indicates the non-dominated solutions whereas the blue colour solutions are indicated by dominated solutions. Out of the all available solutions of non dominants one some of the solutions are considered further analyzed which are near optimal in nature.

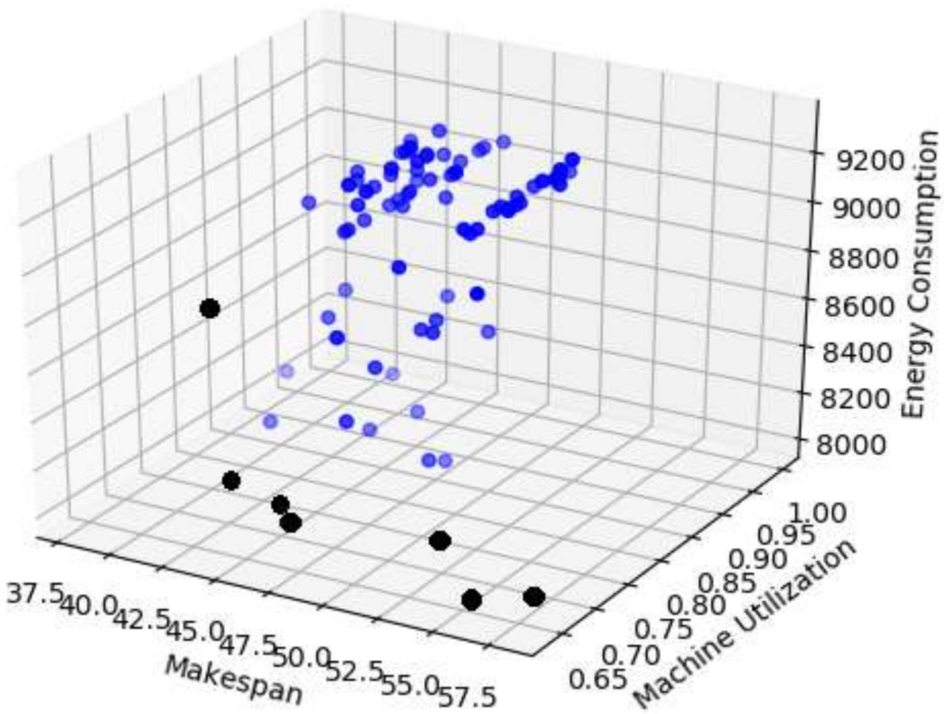


Figure 5.7 Pareto fronts for 6X6 case

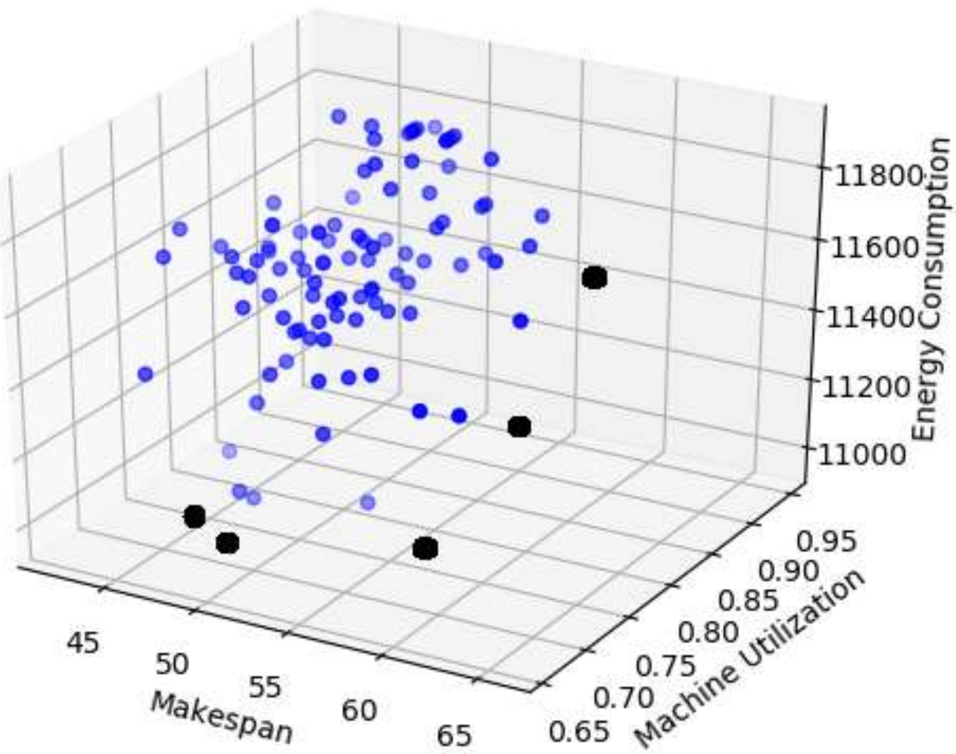


Figure 5.8 Pareto fronts for 8X8 case

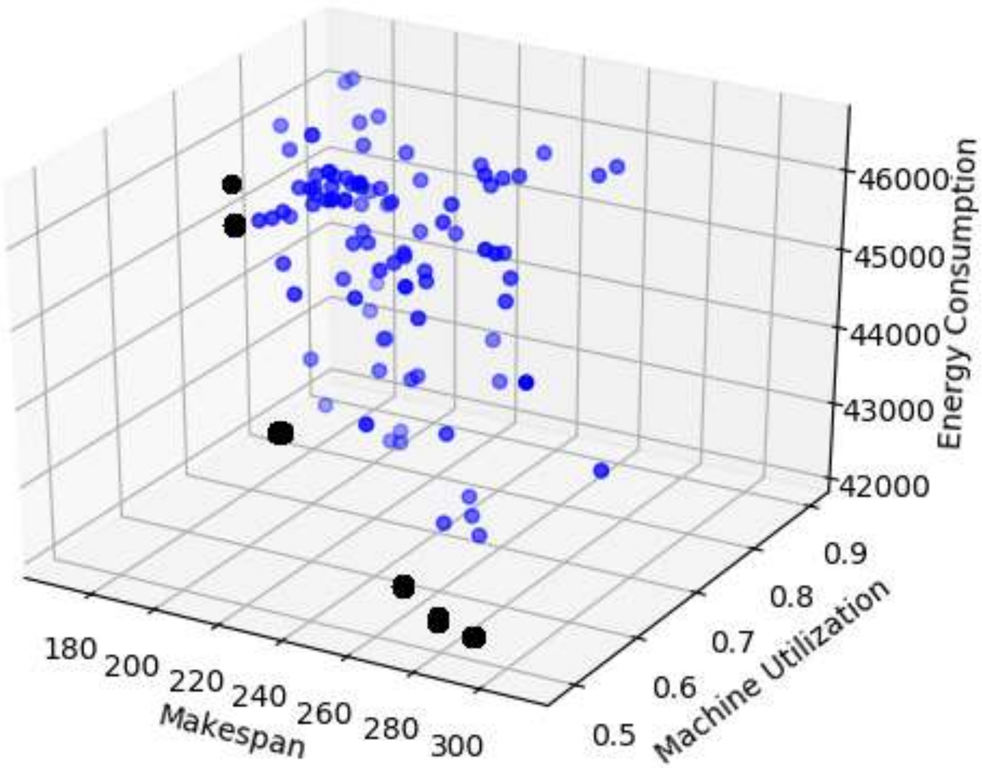


Figure 5.9 Pareto fronts for 6X8 case

This CE-NSGA-II algorithm has been coded in Python 3 and tested on Intel® Core™ i5-4200U CPU @1.6GHz 2.30 GHz, 4GB of RAM. In the end, it can be realized from the results that CE- NSGA-II executes efficiently with a lower number of generations to obtain optimized objective functions with improved convergence and divergence of solutions.

5.9 Conclusive remarks

In the perception of Network-based Manufacturing systems, our work consists of an integrated approach to produce optimal process plans. Investigation of variation between traditional and network manufacturing systems helped us to identify the importance and necessity of networked manufacturing system and their characteristics.

Consequently, problem description with the mathematical model is formulated with certain assumptions, constraints, and makespan minimization, machine utilization maximization, and minimization of energy consumption are taken as objectives with the IPPS approach. Practical machine parameters are characterized like speed, power, etc. and the energy consumption is computed through a mathematical model for each operation-machine combination. The problem being a type of NP-hard complex problem, the CE-NSGA-II algorithm is proposed to fulfill the objective needs and produce a feasible process plan. Further on need, more sub-objectives are introduced like several generations, each job's optimal process plan and computational time which categorize the performance of the proposed algorithm substantially.

Chapter 6

An Integrated blockchain based evolutionary approach for resource sharing and scheduling in a sustainable Networked Manufacturing System

6.1. 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 [113], 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 [114,115].

With the advent of emerging technologies, as well as the importance of individual personalization of customers towards the market [116], 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 [117]. Moreover, the use of external platforms like cloud space [118], 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 [119], 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 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 [120]. More importantly, this technology helps to fulfil the needs of the Digital era [80], 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 [121, 122].

This work 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 work, 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 (HMFO) 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 work 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 work is concluded in Section 8 by providing scope for future work.

6.2. Problem Description

This research work presents a multi-objective MILP model for a sustainable distributed manufacturing system (DMS) shown in Fig. 6.1 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.3.1, 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 [123-125] 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 (HMFO) 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 HMFO. To fulfill the above-stated problem and its objective functions, a mathematical model has been developed and notations of the mathematical model are shown in Table 6.1.

While developing the model few assumptions and constraints are considered that are stated below:

1. Job Pre-emption is not allowed.
2. Before a new job is processed the preceding job must be completed.
3. At an instant processing of more than one job is not possible on the same machine.
4. 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.
5. All the machines are assumed to be ready always.
6. The operations of all the jobs and their sequence must contain future tasks that need to be defined earlier.

Table 6.1. Presents the notations used in the mathematical model.

Notation	Description
Indices	
c	Index of jobs, c ε F
n	Index of alternative process plans, n ε T
a	Index of operations, a ε E
q	Index of machines, q ε R
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 ath 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	
	1 The *nth* alternative process plan of job c is considered
X_{cn}	0 Except above condition
	1 The operation Q_{can} precedes the operation Q_{dbo} on given machine q
$Y_{candboq}$	0 Except above condition
	1 If given machine q is selected for Q_{can}
Z_{canq}	0 Except above condition

Objectives:

Minimization of makespan (W_{min})= $Max\ D_{canq}$ (6.1)

$$\text{Maximization of Machine Utilization } (U_c) = \frac{\sum_{q=1}^{R} H_{cq}}{\sum_{q=1}^{R} (mct_q - mst_q)} \quad (6.2)$$

$$\text{Minimization of energy consumption } (\mathrm{G_{min}}) = \sum_{c=1}^{F} \sum_{a=1}^{E_{cn}} \sum_{q=1}^{R} G_{caq} \quad (6.3)$$

$$\text{Maximization of Reliability } \mathrm{R} = \prod_{o=1}^{v_j} \mathrm{Re}\, l_{jo} \quad (6.4)$$

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

$$D_{cn1q} + \mathrm{O}(1 - X_{cn}) \geq H_{cn1q} \quad (6.5)$$
$$\mathrm{c} \in [1, F],\ \mathrm{n} \in [1, T_C], q \in [1, R]$$

$$D_{cnE_{cn}q} - \mathrm{O}(1 - X_{cn}) \leq H_{cnaq} \quad (6.6)$$
$$\mathrm{c} \in [1, F],\ \mathrm{n} \in [1, T_c], q \in [1, R]$$

$$D_{cnaq} - \mathrm{D}_{cn(a-1)q_1} + \mathrm{O}(1 - X_{cn}) \geq \mathrm{H}_{cnaq} \quad (6.7)$$
$$\mathrm{c} \in [1, F],\ \mathrm{n} \in [1, T_c], a \in [1, E_{cn}], q, q_1 \in [1, R]$$

$$D_{cnaq} - D_{dobq} + \mathrm{O}Y_{cnadobq} \geq \mathrm{H}_{cnaq} \quad (6.8)$$

$$c, d \in [1, F],\ \mathrm{n}, o \in [1, T_c], a, b \in [1, E_{cn}], q \in [1, R]$$

The aforementioned objectives, that is, minimization of makespan, minimization of total training cost of workers, minimization of energy consumption, maximization of service utilization, maximization of reliability is given by Equation (6.1) - (6.4) respectively. Equation (6.5) -(6.8) detailed the constraint related to process plans and precedence relationship of operations.

6.3. 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.6.1, which mainly contains three parts namely, a service layer, a product operational blockchain layer, and a planning layer. In the following sub-sections, we clearly describe the significance of each layer.

6.3.1. 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.

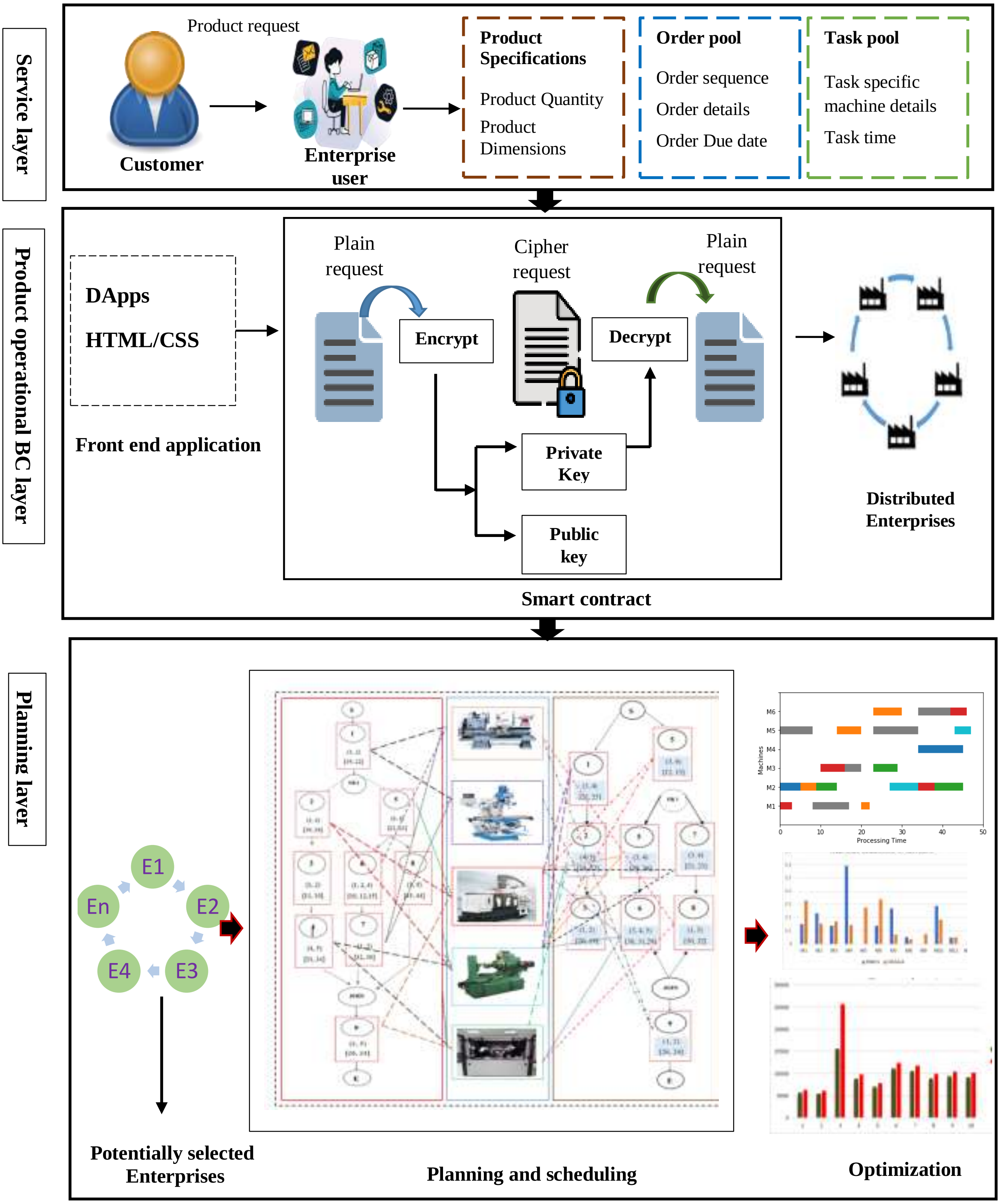


Figure 6.1 Framework of blockchain-assisted evolutionary algorithm approach

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.

6.3.2. 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.

6.3.3. 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.

6.4. Blockchain technology

In distributed manufacturing systems different enterprises are located at various geographical locations are connected through a network. a common database is maintained to share the manufacturing resources to do the process planning and scheduling. This kind of system requires trust between the enterprises to do the transactions. Nowadays after the technological advancements enterprises looks for new technology which can help them to eliminate the demerits caused by trusting each other in the global competitive market. In the phase a new emerging technology called Block Chain changed the scenario of trust-based resource sharing and economic activities.

Blockchain is a technology that allows data to be stored and exchanged on a peer-to-peer basis. Structurally, blockchain data can be consulted, shared and secured. It is used in a decentralized manner and removes the need for intermediaries, or "trusted third parties". Authentication of transactions is achieved through cryptographic means and a mathematical “consensus protocol” that determines the rules by which the ledger is updated, which allows participants with no particular trust in each other to collaborate without having to rely on a single trusted third party. Participants in a blockchain can access and check the ledger at any time. Blockchain therefore ensures immediate, across-the-board transparency, and as transactions added to the blockchain are time-stamped and cannot easily be tampered with, blockchain technology allows products and transactions to be traced easily.

Each block in the block chain consists of time stamp and the hash and a nonce which makes a block chain special from other technologies. Hash values of Previous $(I-1)^{th}$ block is connected to the present block (I^{th}) block as shown in Fig.6.2. All the hash values of the blocks connected to parent block and modifications in the any block leads to immediately change in the hash value of all other blocks which makes the block chain as immutable. As mentioned above any information added to the blockchain will be notified to all the other nodes in the block chain. Another important issue that makes the need for block chain in distributed manufacturing industries is that to avoid un ethical tempering of data takes place by tracing back for its verification in the block chain.

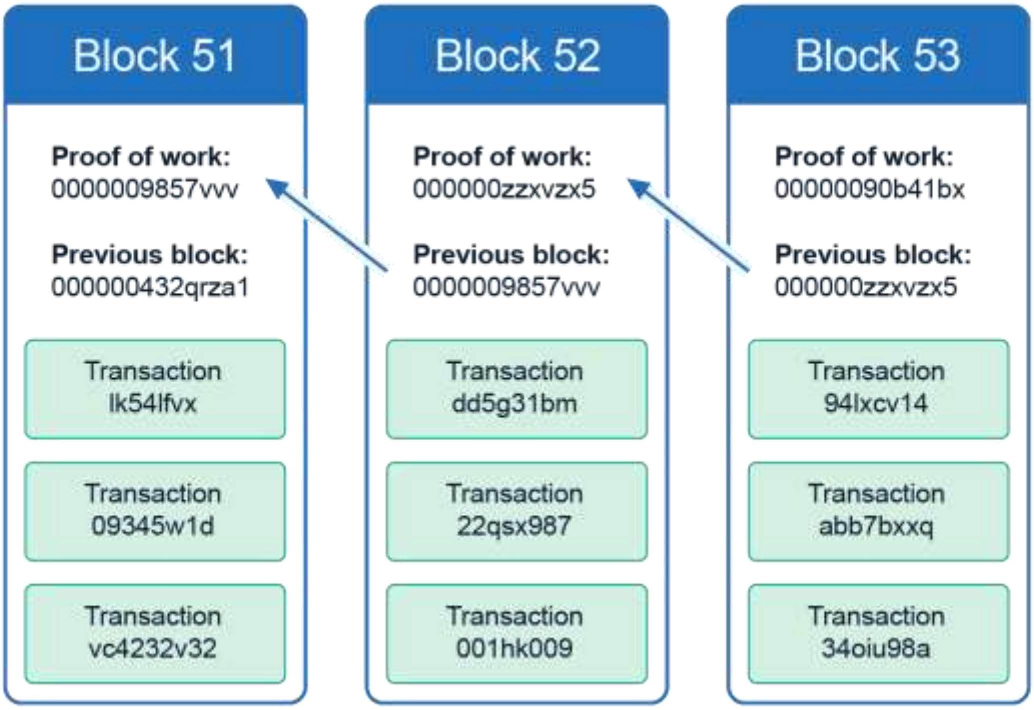


Figure 6.2 Block chain structure

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.

Authentication of transactions is achieved through cryptographic means and a mathematical "consensus protocol" that determines the rules by which the ledger is updated, which allows participants with no particular trust in each other without depending on a single trusted third party. Blockchain therefore ensures immediate, across-the-board transparency, and as transactions added to the blockchain are time-stamped and cannot easily be tampered with, blockchain technology allows for block chain in distributed manufacturing industries.

6.4.1. 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 research work, 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. 6.1 shows the blockchain framework for DMS. All the participants must possess an account that has a specific address that is nothing but the user public key.

6.4.2. 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.

6.4.3. Implementation of the proposed blockchain model in Ethereum

The proposed BC model includes two smart contracts. The first smart contract in Table 6.2 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 6.3, 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. 6.3 denotes the proposed model for working with smart contracts in the BC.

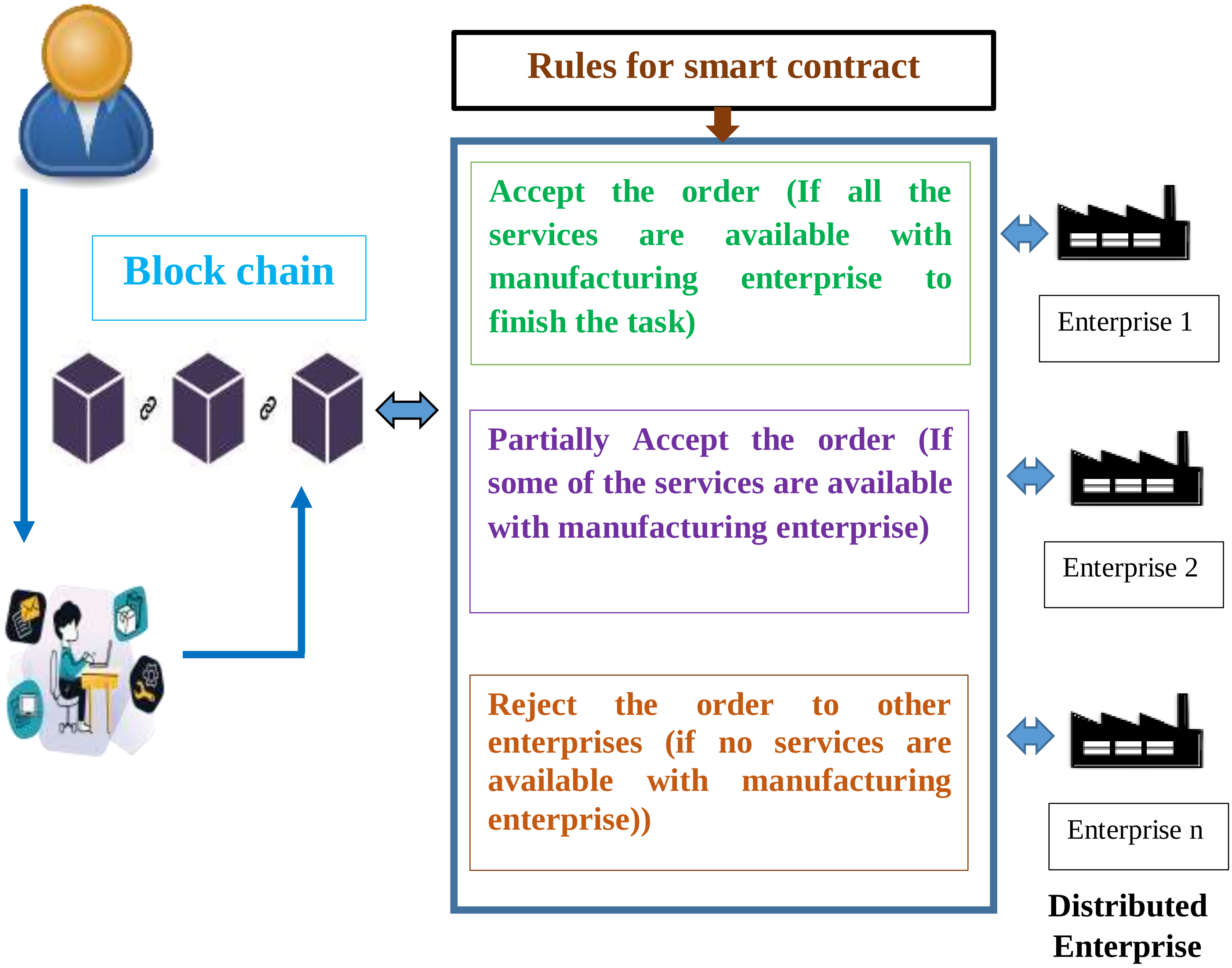


Figure 6.3 Proposed model diagram for working of smart contract in the blockchain

6.5 Results of Blockchain based smart contracts

Table 6.2 shows the first smart contract shown in 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-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. The screenshot of first smart contract written in the Ethereum platform shown in the Fig. 6.4.

Table 6.2. 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 6.3 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 is 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 x and the enterprise offers that service x, 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";*

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.7.4;

contract ordering
{
    //structure for placing order
    struct order
    {
        string name;
        uint quantity;
        string color;
        string expected_date;
    }

    uint order_seq;
    mapping(uint => order) orders;

    function create_order(string memory name,uint quantity,string memory color,string memory expected_date) public
    {
        order_seq++;
        orders[order_seq] = order(name,quantity,color,expected_date);
    }

    function queryOrder(uint number) view public returns (string memory name,uint quantity,string memory color,string
    {
        return(orders[number].name,orders[number].quantity,orders[number].color,orders[number].expected_date);
    }

    function total_orders() view public returns (uint)
    {
        return order_seq;
```

Figure 6.4 shows the screenshot of smart contract between customers and enterprises in Ethereum platform

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.7.4;
import "./ordering.sol";

contract distribution
{
    address public enterprise1 = 0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2;
    address public enterprise2 = 0x4B20993Bc481177ec7E8f571ceCaE8A9e22C02db;
    address public enterprise3 = 0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB;
    address public enterprise4 = 0x617F2E2fD72FD9D5503197092aC168c91465E7f2;
    address public enterprise5 = 0x17F6AD8Ef982297579C203069C1DbfFE4348c372;
    address public enterprise6 = 0x5c6B0f7Bf3E7ce046039Bd8FABdfD3f9F5021678;
    address public enterprise7 = 0x03C6FcED478cBbC9a4FAB34eF9f40767739D1Ff7;
    address public enterprise8 = 0x1aE0EA34a72D944a8C7603FfB3eC30a6669E454C;
    address public enterprise9 = 0x0A098Eda01Ce92ff4A4CCb7A4fFFb5A43EBC70DC;
    address public enterprise10 = 0xCA35b7d915458EF540aDe6068dFe2F44E8fa733c;



    event status_of_enterprise(string s);
    //structure for placing order
    /*struct order
    {
        string name;
        uint quantity;
        string color;
        string expected_date;
    }*/

    //structure for enterprise and the services it offers
    struct enterprise
```

Figure 6.5 shows the smart contract between enterprise user and distributed enterprise in the block chain.

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. The screenshot of first smart contract written in the Ethereum platform shown in the Fig. 6.5. For testing, each enterprise is assigned with particular Ethereum addresses and Fig. 6.6 indicates the various addresses that are created for Enterprise 1 and Enterprise 2.

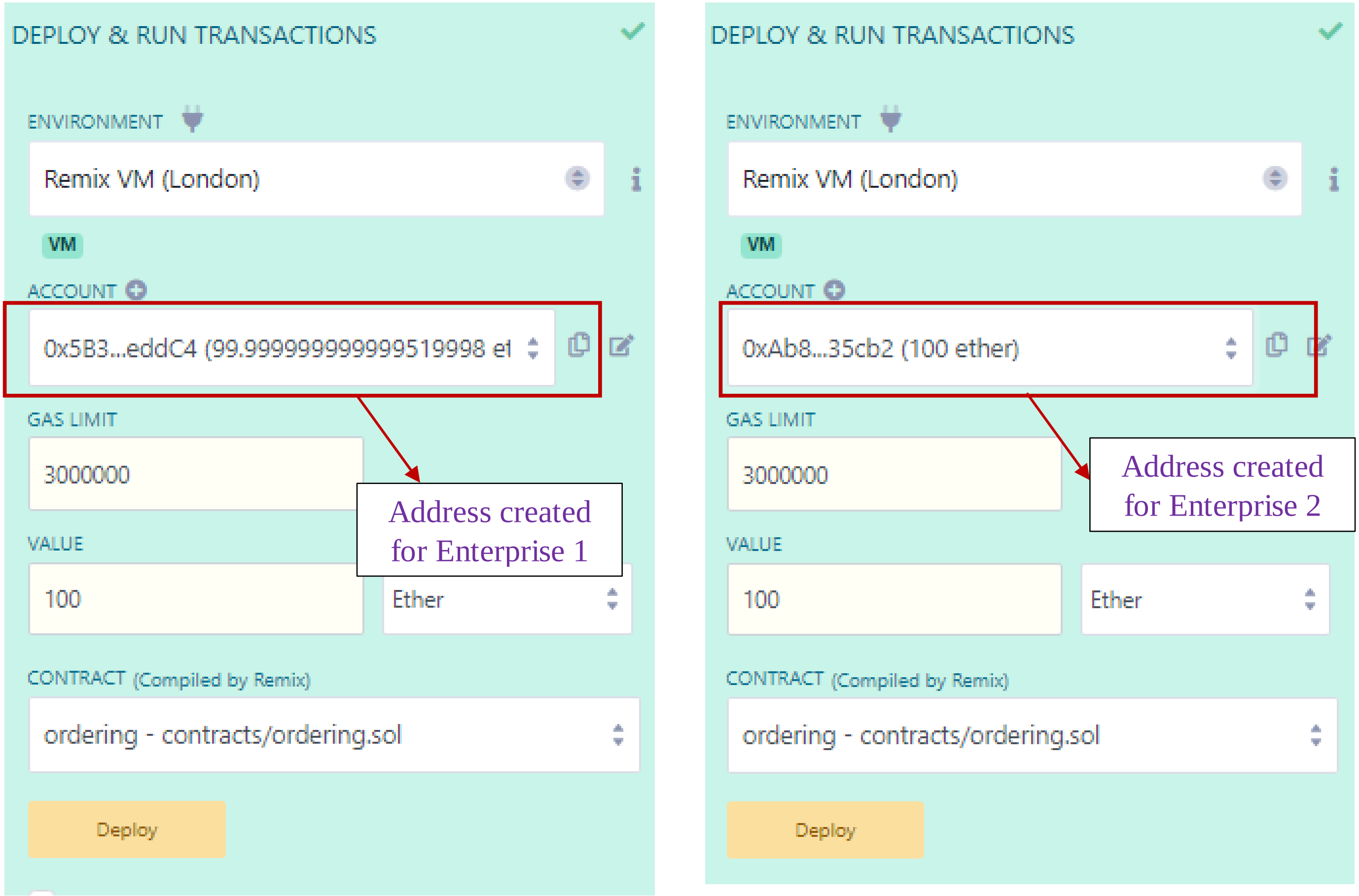


Figure 6.6 Various addresses are created for Enterprises 1 and 2

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 Fig. 6.7 to Fig. 6.16.

```
[vm] from: 0xAb8...35cb2 to: distribution.check enterprise (uint256) 0xd91...39138 value: 0
wei data: 0x163...00001 logs: 1 hash: 0xaaa...9677e

Status              true Transaction mined and execution succeed

transaction hash    exaaaf1f253f9ac9d268b77bd411b45524e04fa57ee67402c32c3e11a90e09677e

from                0xCA35b7d915458EF540aDe6068dFe2F44E8fa733c

to                  distribution.check enterprise exd9145CCE520386f254917e481e844e9943F39138

gas                 100000000 gas

transaction cost    55017 gas

execution cost      33553 gas

hash                exaaaf1f253f9ac9d268b77bd411b45524e04fa57ee67402c32c3e11a99e09677e

input               0x163...00001

decoded input       {"uint256 id": { "type": "BigNumber", "hex": "0x01" } }

decoded output      {"0": "string: stat PARTIALLY ACCEPT"}

logs                [ { "from": "0xd9145CCE52D386f254917e481e844e9943F39138", "topic":
                   "0xab296d786472fbbe1dcebd72681ffa65fd89a47fe095f95e84f43c385961b235",
                   "event": "status of enterprise", "args": {"0":
                    "PARTIALLY ACCEPT", "s": "PARTIALLY ACCEPT"}}]
Value               Ø wei L
```

Figure 6.7 Transactions recorded on the Ethereum based blockchain for case I- E1

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 6.7 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 Partially Accept the order highlighted in Fig. 6.7.

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. 6.8.

```
[vm] from: 0x4B2...C02db to: distribution.check enterprise (uint256) 0xd91...39138 value: 0
wei data: 0x163...00002 logs: 1 hash: 0xedc...4f3a2

Status               true Transaction mined and execution succeed

transaction hash     exedcd0ebdc86f87271ab74f927ff2b35530c0f9b328c1980b689cd2864a44f3a2

from                 ex45209938c481177ec7E8f571ceCaE8A9e22C02db

to                   distribution.check enterprise exd9145CCE520386f254917e481e844e9943F39138

gas                  100000000 gas

transaction cost     56212 gas

execution cost       34748 gas

hash                 exedcd0ebdc86f87271ab74f927ff2b35530c0f9b328c1980b689cd2064a44f3a2

input                0x163...00002

decoded input        {"uint256 id": { "type": "BigNumber", "hex": "0x82" } }

decoded output       {"0": "string: stat ACCEPT"}

logs                 [ { "from": "0xd9145CCE52D386f254917e481e844e9943F39138", "topic":
                    "0xab296d786472fbbe1dcebd72681ffa65fd89a47fe095f95e84f43c385961b235",
                    "event": "status of enterprise", "args": {"0":

                     "ACCEPT", "s": "ACCEPT"}}]

Value                Ø wei L
```

Figure 6.8 Transactions recorded on the Ethereum based blockchain for case II- E2

```
[vm] from: 0x787...cabaB to: distribution.check enterprise (uint256) 0xd91...39138 value: wei
data: 0x163...00003 logs: 1 hash: 0x1df...103c8

Status               true Transaction mined and execution succeed

transaction hash     0x1df746638ecb8b6f1757a71669369510583f7f7933b8ab9d56f68b3780e103c8

from                 0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB

to                   distribution.check enterprise exd9145CCE520386f254917e481e844e9943F39138

gas                  100000000 gas

transaction cost     56827 gas

execution cost       35363 gas

hash                 0x1df746638ecb8b6f1757a71669369510583f7f7933b8ab9d56f68b3780e103c8

input                0x163...00003

decoded input        {"uint256 id": { "type": "BigNumber", "hex": "0x03" } }

decoded output       {"0": "string: stat REJECT"}

logs                 [{ "from": "0xd9145CCE52D386f254917e481e844e9943F39138", "topic":
                    "@xab296d786472fbbe1dcebd72681ffa65fd89a47fe095f95e84f43c385961b235",
                    "event": "status of enterprise", "args": {"0":

                     "REJECT",

                     " REJECT ", "s": " REJECT "}}]

Value                Ø wei L
```

Figure 6.9 Transactions recorded on the Ethereum based blockchain for case III- E3.

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. 6.9.

```
[vm] from: 0x617...5E7f2 to: distribution.check_enterprise(uint256) 0xd91...39138 value: wei
data: 0x163...00004 logs: 1 hash: 0x389...a0e79

Status               true Transaction mined and execution succeed
transaction hash     ex38924e2b573dec09bc8d6aa17f803fd1d15c8ef39af7ea094fb3064cd@da0e79
from                 0x617F2E2FD72FD9D5503197092aC168c91465E7f2
to                   distribution.check_enterprise 0xd9145CCE52D386f254917e481e844e9943F39138
gas                  100000000 gas
transaction cost     57503 gas
execution cost       36039 gas
hash                 ex38924e2b573dec09bc8d6aa17f803fd1d15c8ef39af7ea094fb3064cd@da0e79
input                0x163...00004
decoded input        {"uint256 id": { "type": "BigNumber", "hex": "0x84" } }
decoded output       {"0": "string: stat REJECT"}
logs                 [{ "from": "0xd9145CCE52D386f254917e481e844e9943F39138", "topic":
                    "0xab296d706472fbbe1dcebd72681ffa65fd89a47fe095f95e84f43c385961b235",
                    "event": "status_of_enterprise", "args": {"0":
                     "REJECT",
                     " REJECT ", "s": " REJECT "}}]
Value                Ø wei
```

Figure 6.10 Shows transactions recorded on the Ethereum block chain for case IV- E4 interacts with the blockchain.

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. 6.10. From the Fig. 6.10 an important observation can be noted that once the smart contract rejects the order or a particular enterprise E4 immediately the smart contract checks the availability with the other enterprises. In this way block chain based smart contract helps to identify the suitable enterprise that have the capability to offer the service to make the product. Moreover, entire transaction has completed through the block chain and it will be stored in the blockchain. There is no scope for the enterprise to deny to offer service. The immutable property of blockchain makes this achievable due to the nature of the BC that the data enter into the blockchain cannot be tempered or modified.

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 6.11.

```
[vm] from:0x17F...8c372 to: distribution.check_enterprise (uint256) 0xd91...39138 value:
0 wei data: 0x163...00005 logs: 1 hash: 0x945...40f05

Status              true Transaction mined and execution succeed
transaction hash    8x945c184bca2231ff76d9a41d2452ad80a0fed77ee7128d5fa5343e3e78f40f05
from                8x17F6AD8EF982297579C203069C1DbfFE4348c372
to                  DCE (uint256) 0xd9145CCE52D386f254917e481e44e9943F39138
gas                 100000000 gas
transaction cost    58513 gas
execution cost      37049 gas
hash                0x945c184bca2231ff76d9a41d2452ad80a0fed77ee7128d5fa5343e3e78f40f05
input               0x163...00005
decoded input       {"uint256 id": { "type": "BigNumber", "hex": "0x05" } }
decoded output      {"0": "string: stat PARTIALLY ACCEPT "}
logs                [ { "from": "0xd9145CCE52D386f254917e481e644e9943F39138","topic":
                    "exab296d706472fbbeldcebd72681ffa65fd89a47fe095f95e84f43c385961b235",
                    "event": "status_of_enterprise", "args": {"0":
                    "PARTIALLY ACCEPT", "s": "PARTIALLY ACCEPT"}}]
Value                ø wei
```

Figure 6.11 Shows transactions recorded on the Ethereum block chain for case V- E5 interacts with the blockchain.

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 6.12.

```
[vm] from: 0x5c6...21678 to: distribution.check_enterprise (uint256) 0xd91...39138
value: 0 wei data: 0x163...00006 logs: 1 hash: 0xf1e...ba591

Status              true Transaction mined and execution succeed

transaction hash    exf1e0243cc0d5ff2bea67355f8dc2386894eb4b9df22db9f635731bdf01dba591

from                0x5c680f7Bf3E7ce046039Bd8FABdfD3f9F5021678

to                  DCE (uint256) 0xd9145CCE520386f254917e481e844e9943F39138

gas                 100000000 gas

transaction cost    59472 gas

execution cost      38008 gas

hash                0xf1e0243cc0d5ff2bea67355f8dc2386894eb4b9df22db9f635731bdf01dba591

input               0x163...00006

decoded input       {"uint256 id": { "type": "BigNumber", "hex": "0x06" } }

decoded output      {"0": "string: stat PARTIALLY ACCEPT "}

logs                [ { "from": "0xd9145CCE52D386f254917e481e644e9943F39138","topic":

                    "exab296d706472fbbeldcebd72681ffa65fd89a47fe095f95e84f43c385961b235",
                    "event": "status_of_enterprise", "args": {"0":

                    "PARTIALLY ACCEPT", "s": "PARTIALLY ACCEPT"}}]

Value                Ø wei
```

Figure 6.12 shows transactions recorded on the Ethereum block chain for case VI- E6 interacts with the blockchain.

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 6.13.

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 6.14.

```
[[vm] from: 0x03C...D1FF7 to: distribution.check_enterprise (uint256) 0xd91...39138
value: 0 wei data: 0x163...00007 logs: 1 hash: 0x946...96316

Status              true Transaction mined and execution succeed

transaction hash    0x946640fa761490d9c8f53c6bb6ae64815607e6cb147d5aaa858517c1f5496316

from                0x03C6FcED478cBbC9a4FAB34F9f40767739D1Ff7

to                  DCE (uint256) 0xd9145CCE52D386f254917e481e644e9943F39138

gas                 100000000 gas

transaction cost    60582  gas

execution cost      39118 gas

hash                0x946640fa761490d9c8f53c6bb6ae64815607e6cb147d5aaa858517c1f5496316

input               ex163...00007

decoded input       {"uint256 id": { "type": "BigNumber", "hex": "0x07" } }

decoded output      {"0": "string: stat ACCEPT "}

logs                [ { "from": "0xd9145CCE52D386f254917e481e644e9943F39138","topic":

                     "@xab296d706472fbbeldcebd72681ffa65fd89a47fe095f95e84f43c385961b235",
                     "event": "status_of_enterprise", "args": {"0":

                    "ACCEPT", "s": "ACCEPT"}}]

Value                ø wei
```

Figure 6.13 shows transactions recorded on the Ethereum block chain for case VII- E7 interacts with the blockchain.

```
[vm] from: 0x1aE...E454C to: distribution.check_enterprise (uint256) 0xd91...39138
value: 0 wei data: 0x163...00008 logs: 1 hash: 0xa3c...13433

Status              true Transaction mined and execution succeed

transaction hash    Oxa3c1a6b5436fd3f1bfa0233512523ed38d920bf0604d994c118126be34413433

from                0x1aE0EA34a72D944a8C7603FFB3eC30a6669E454C

to                  DCE (uint256) 0xd9145CCE52D386f254917e481e44e9943F39138

gas                 100000000 gas

transaction cost    61494 gas

execution cost      40030 gas

hash                exa3c1a6b5436fd3f1bfa0233512523ed38d920bf0604d994c118126be34413433

input               1 0x163...00008

decoded input       {"uint256 id": { "type": "BigNumber", "hex": "0x08" } }

decoded output      {"0": "string: stat PARTIALLY ACCEPT "}

logs                [ { "from": "0xd9145CCE52D386f254917e481e644e9943F39138","topic":

                    @xab296d706472fbbeldcebd72681ffa65fd89a47fe095f95e84f43c385961b235",
                    "event": "status_of_enterprise", "args": {"0":

                    "PARTIALLY ACCEPT", "s": "PARTIALLY ACCEPT"}}]

Value                ø wei
```

Figure 6.14 shows transactions recorded on the Ethereum block chain for case VIII- E8 interacts with the blockchain.

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 services available to perform the task. And the corresponding result is mentioned in Fig 6.15.


```
[vm] from: 0x0A0...C70DC to: distribution.check_enterprise(uint256) 0xd91...39138
value: 0 wei data: 0x163...00009 logs: 1 hash: 0xda3...34ce9

Status               true Transaction mined and execution succeed

transaction hash     exda3a8ec8acf8c636487f417feb041c0fb7de904562c32c1c0dab578cd6234ce9

from                 0x0A098Eda01Ce92ff4A4CCb7A4FFFB5A43EBC70DC

to                   DCE (uint256) 0xd9145CCE52D386f254917e481e844e9943F39138

gas                  100000000 gas

transaction cost     61939 gas

execution cost       40475 gas

hash                 exda3a8ec8acf8c636487f417feb041c0fb7de904562c32c1c0dab578cd6234ce9

input                1 0x163...00009

decoded input        {"uint256 id": { "type": "BigNumber", "hex": "0x09" } }

decoded output       {"0": "string: stat PARTIALLY REJECT "}

logs                 [ { "from": "0xd9145CCE52D386f254917e481e844e9943F39138", "topic":
                     "0xab296d706472fbbeldcebd72681ffa65fd89a47fe095f95e84f43c385961b235",
                     "event": "status_of_enterprise", "args": {"0":

                     "REJECT", "s": "REJECT" }}]


Value                 Ø wei
```


Figure 6.15 Shows transactions recorded on the Ethereum block chain for case IX- E9 interacts with the blockchain.

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. 6.16.

Figure 6.16 shows transactions recorded on the Ethereum block chain for case X- E10 interacts with the blockchain.

From the Fig. 6.16 an important observation can be noted that once the smart contract partially rejects the order for a particular enterprise E10 immediately the smart contract checks the availability with the other enterprises to fulfill the other services. In this way block chain based smart contract helps to identify the suitable enterprise that have the capability to offer the service to make the product. Moreover, entire transaction has completed through the block chain and it will be stored in the blockchain. There is no scope for the enterprise to deny to offer service. The immutable property of blockchain makes this achievable due to the nature of the BC that the data enter into the blockchain cannot be tempered or modified. From the Fig. 6.17 screenshot of all the transactions in the Ethereum blockchain are shown.

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 @ 1.80GHz, 32 GB RAM Ubuntu 18.04 LTS. Remix Ethereum- pragma solidity ^0.7.4 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.

Figure 6.17 Screen shot of transactions recorded on the Ethereum blockchain for various cases

. Apart from the smart contract helps to identify the right enterprise that has the capability to offer the services required to manufacturing the product. 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. 6.1. 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

6.6. Multi-Objective Hybridized Moth Flame Optimization (HMFO) Algorithms

To solve the considered problem a newly established Bio-inspired Moth Flame Optimization algorithm (MFO) has been adopted and further it has been mapped according to problem nature [101]. The superiority of MFOA over other algorithms Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO) is clearly shown in his work by conducting tests on several benchmark functions [102]. In this present work, a hybridized form of Moth Flame Optimization algorithm (HMFO) 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 HMFO algorithm results were compared with a reference point based multi-objective algorithm NSGA-III is proposed by [126] 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.

Table 6.4 Initialization of parameters for proposed solution algorithm.

Process Parameters	HMFO	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

The parameters for the HMFO technique are specified for the implementation of algorithms shown in Table 6.4 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.

Working of HMFO to the current problem of NMS.

Step 1: In HMFO 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.

$$K = \begin{bmatrix} K_{11} & K_{12} & .... & K_{1d} \\ K_{21} & K_{22} & .... & K_{2d} \\ K_{31} & K_{32} & .... & K_{3d} \\ K_{41} & K_{42} & .... & K_{4d} \end{bmatrix} \qquad (6.9)$$

Initialization of moth population and their spaces are defined with the time matrices and their corresponding inputs. In this proposed HMFO a new type of encoding schema was presented to suit the problem nature. The encoding scheme for makespan is presented in Fig. 6.18.

c) Chromosome encoded based on processing time of each operation on corresponding machines

6	**14**	**11**	**16**	**8**	**13**	**7**	**25**	**6**
Machine (M_1)			**Machine (M_2)**			**Machine (M_3)**		

b) Chromosome encoded based on selection of machines

2	**1**	**3**	**3**	**1**	**2**	**1**	**2**	**3**
Operation (O_1)			**Operation (O_2)**			**Operation (O_3)**		

a) Chromosome encoded based on sequence of operation

3	**2**	**1**	**2**	**3**	**1**	**1**	**3**	**2**
Job (J_1)			**Job (J_2)**			**Job (J_3)**		

Figure 6.18 Representation of chromosome initialization for make span

To understand the encoding schema in Fig. 6.18 mainly contains three jobs that are to be processed on three machines and, each machine demands three operations. Furthermore, specifically observe the Fig. 6.18 a) encoding explains the number of operations and their sequence to be followed for each job i.e. job1, job2 and, job3 whereas Fig. 6.18 b) detailed the particular operations and their corresponding available machine for processing. Similarly, in Fig. 6.18 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 particular operation for the xth job and yth process plan. The remaining values in the matrix are kept as zeros.

	M1	**M2**	**M3**	**M4**	**M5**	**M6**	**M7**	**M8**	**M9**	**M10**	**M11**	**M12**
O_1	[6	16	7	0	0	0	0	0	0	0	0	0;
O_2	14	8	25	0	0	0	0	0	0	0	0	0;
O_3	11	13	16	0	0	0	0	0	0	0	0	0;
O_4	0	0	0	0	0	0	0	0	0	0	0	0;
O_5	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 6.5 leads to the energy consumption matrix (Equation (6.10)). The Fig. 19 indicates the corresponding encoding schema.

Energy consumption matrix (o, m1, p1, j) = Makespan matrix (o, m, p, j) *Energy (Rated energy matrix); (6.10)

To understand the encoding schema in Fig. 6.19, upon careful observation Fig. 6.19 a) encoding explains the number of operations and their sequence to be followed for each job i.e. job1, job2 and, job3 whereas Fig. 6.19 b) detailed the particular operations and their corresponding available machine for processing. Similarly, in Fig.6.19 c) explains the encoding schema for energy consumption of corresponding machine for a given operation. The above matrix indicating the energy consumption values of machines for the corresponding operation for the 3rd job and 2ndprocess plan. The remaining values in the matrix are kept as zeros.

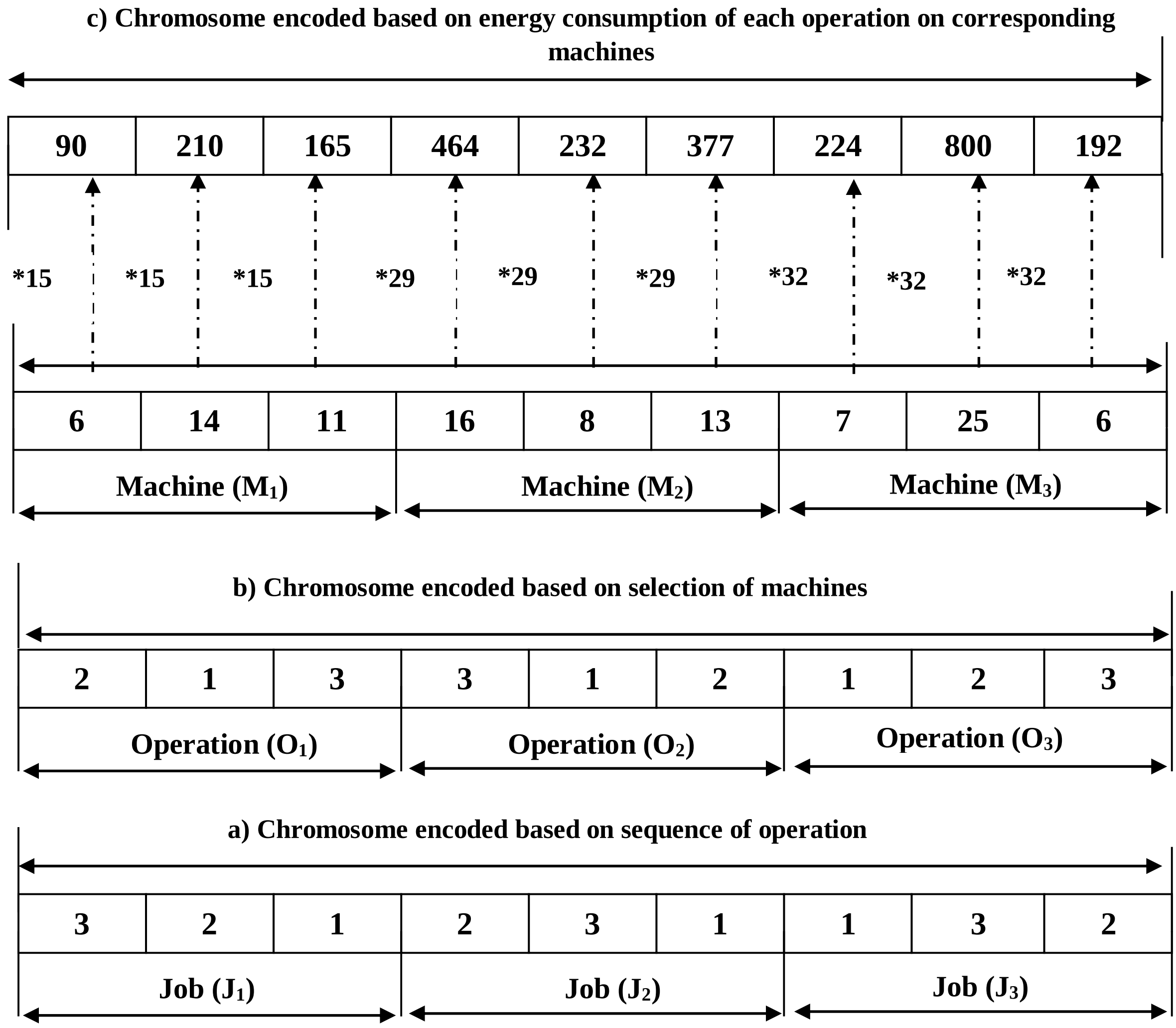


Figure 6.19 Representation of chromosome initialization for energy consumption

	M1	**M2**	**M3**	**M4**	**M5**	**M6**	**M7**	**M8**	**M9**	**M10**	**M11**	**M12**
O_1	[90	464	224	0	0	0	0	0	0	0	0	0;
O_2	210	232	800	0	0	0	0	0	0	0	0	0;
O_3	165	377	192	0	0	0	0	0	0	0	0	0;
O_4	0	0	0	0	0	0	0	0	0	0	0	0;
O_5	0	0	0	0	0	0	0	0	0	0	0	0];

Step 3 The selection of process plan out of available process plans is carried by score function shown in Equation (6.11), a lower score value leads to a selection of better process plan.

$$\text{Score} = \frac{\text{Makespan X Energy consumption}}{\text{Reliability}} \tag{6.11}$$

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

$$\text{FK} = \begin{bmatrix} FK_1 & FK_2 & FK_3 & FK_4 \end{bmatrix}^T$$

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 (6.12)

$$Z\ (K_x, L_y) = S_x \times e^{at} \cos(2\pi t) + L_y. \tag{6.12}$$

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, $S_x = \| L_y - K_x \|$, 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 0s, 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.

Table 6.5 Energy and reliability data.

Machine ID	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12
Energy consumption (KJ)	22	26	36	15	12	15	16	31	12	23	27	18
Reliability	0.76	0.82	0.78	0.84	0.84	0.92	0.89	0.94	0.88	0.95	0.84	0.92

The flowchart for the proposed HMFO is presented in Fig.4.3. 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.

6.7. Discussion and Results

The efficiency of the HMFO 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 6.5 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.

Table 6.6 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 (KJ)	MU	CPU Time (Sec)	Makepsan (Minutes)	EC (KJ)	MU	CPU Time (Sec)
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

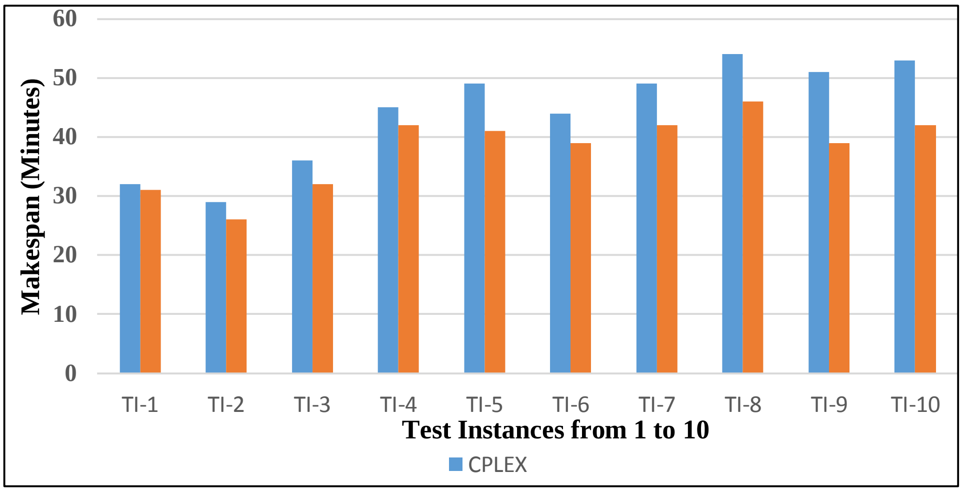


Figure 6.20 Comparison of Makespan values of all Test Instances 1 to 10 (TI- 1 to TI 10) for CPLEX and HMFO.

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 seconds was taken into account for solving the test scenarios [114] mentioned in Table 6.6. A comparison of cplex results with the proposed HMFO was shown in Table 6.6. All the three objectives namely makespan, energy consumption, machine utilization values obtained by an augmented e- constraint method followed in CPLEX solver of GAMS [115] and the results are compared with the objective values of proposed HMFO.

The lower makespan indicating the superiority of proposed algorithm HMFO over the other algorithm for the considered test instances is shown in the Fig. 6.20. The X axis representing the test instances and Y axis represents the makespan values (minutes) shown in Fig. 6.20. The lower energy consumption and higher machine utilization values of proposed HMFO for all the test instances indicates the better performance of the algorithm over all the test scenarios shown in Fig. 6.21 and Fig.6.22. In figures 6.20, 6.21 and, 6.22 shown the CPLEX results are indicated with blue colour and HMFO results are indicated with orange colour. 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.

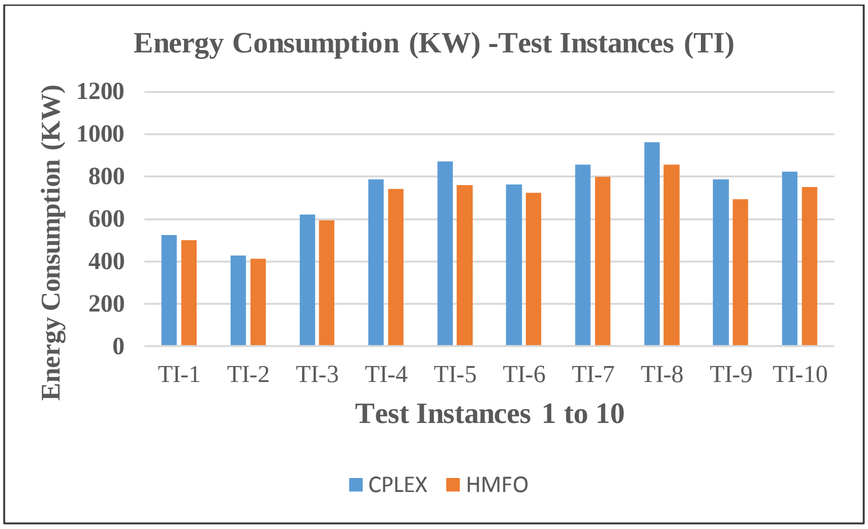


Figure 6.21 Comparison of Energy Consumption values of all Test Instances 1 to 10 (TI 1 to 10) for CPLEX and HMFO.

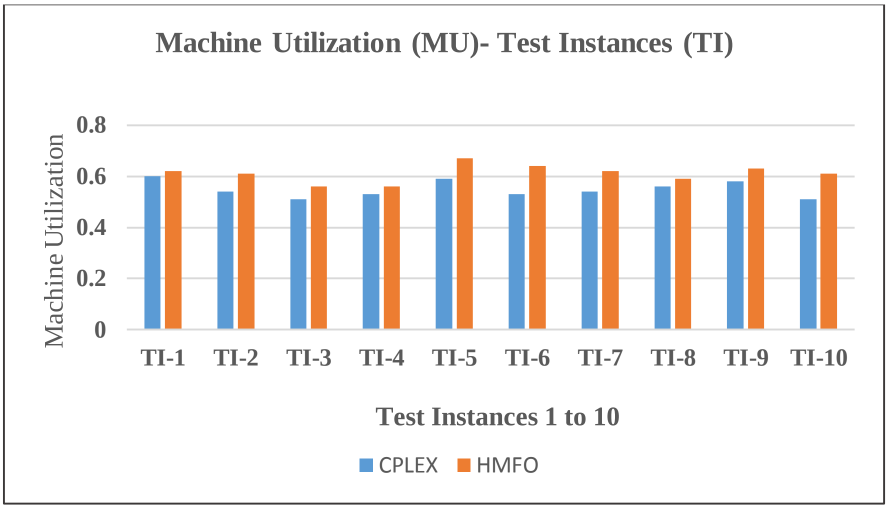


Figure 6.22 Comparison of Machine Utilization values of all Test Instances 1 to 10 (TI 1 to 10) for CPLEX and HMFO.

6.7.1 Comparison of the considered HMFO with the experimental scenarios.

To check the feasibility of the proposed HMFO method a comparison study has been carried out in this work by considering the experimental scenarios 1 to 36 mentioned in Table 6.7. From Table 6.7 scenarios 1 to 32 mentioned by [46] used the Genetic Algorithm based Simulated Annealing (GA-SA) were considered. Whereas in scenarios 33 to 35 mentioned by [47] 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 HMFO for all the practical scenarios 1 to 36 and their values are shown in Table 6.7. Furthermore, the superiority of the proposed algorithm for all the experimental instances were identified.

Table 6.7 Comparison of make span and energy consumption results for all experimental scenarios.

	Jobs	Machines	GA-SA (Scenario 1 to 32)		Proposed HMFO	
			Makespan (Mniutes)	**EC (KJ)**	**Makespan (Mniutes)**	**EC (KJ)**
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**	**72693**
Scenario31	Twenty	Twenty	7753	100073.4	**6356**	**85741**
Scenario32	Twenty	Twenty	15062	197787.5	**13897**	**173652**
Scenario 33	Eighteen	Fifteen	531	13340.3	**523**	**12869**
Scenario 34	Eighteen	Fifteen	810	2036.32	**719**	**1895**
Scenario 35	Eighteen	Fifteen	680	2267.88	**582**	**1742**

6.7.2 Comparison of the considered HMFO with the practical scenarios.

Hereafter thorough analysis of experimental scenarios (Table 6.7) and, the effectiveness of the proposed HMFO algorithm. We further considered the Practical scenarios 1 to 10 in Table 6.8. 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 6.8. Moreover, Table 6.8 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 select the optimal process plan out of all available process plans for each job. Both the proposed HMFO 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 6.9 with their Pareto-optimal values of makespan and energy consumption.

Table 6.8 Optimal process plans selected for each job for all scenarios 1 to 10.

Scenario	Different Cases		Chosen Process plans							
	Jobs	Machines	Job1	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	-	-

From Table 6.9, 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 HMFO 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 HMFO and NSGA III, respectively. From this, we may conclude that the proposed HMFO 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 [53]. Upon observing and comparison of both proposed HMFO and NSGA III for all the scenarios i.e (1 to 10) the HMFO algorithm obtained far better Pareto-optimal results when compared to the standard NSGA III algorithm.

Table 6.9 Results of the practical scenarios with makespan and energy consumption values

Scenario	Number of Jobs	Machines	Proposed HMFO		NSGA III	
			Makespan (Mniutes)	EC (KJ)	Makespan (Mniutes)	EC (KJ)
Scenario 1	six	six	51	5722.29	55	6358
Scenario 2	six	six	48	5481	62	6090
Scenario 3	six	Eight	160	15654	187	25636
Scenario 4	Eight	Eight	59	8846	69	9828
Scenario 5	Eight	Eight	68	7050	82	7833
Scenario 6	six	Twelve	693	11148	793	12386
Scenario 7	six	Twelve	1023	10575	1153	11750
Scenario 8	six	Twelve	1398	8950	1479	9944
Scenario 9	six	Twelve	2050	9387	2195	10429
Scenario10	six	Twelve	1310	9183	1560	10203

To understand the results obtained by the HMFO algorithm in a 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. 6.23 to Fig. 6.27. All remaining scenarios, i.e. from scenarios 6 to 10, are represented as Gantt charts in Fig. 6.28 to 6.32.

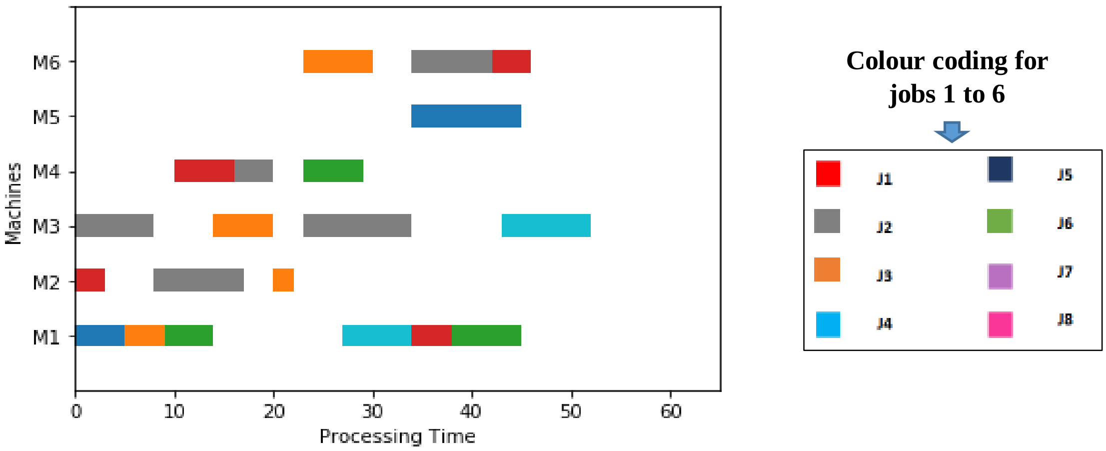


Figure 6.23 Gantt charts for the scenario 1 for the proposed HMFO

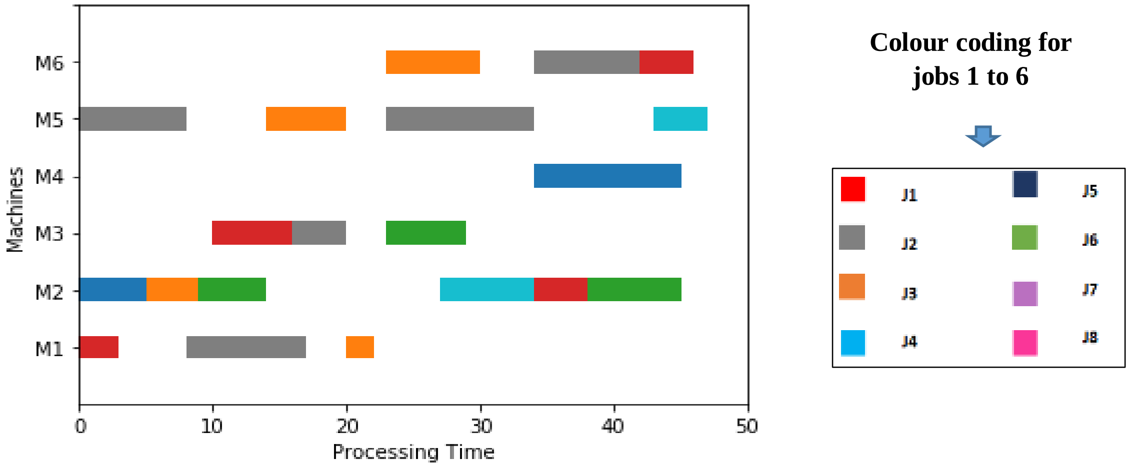


Figure 6.24 Gantt charts for the scenario 2 for the proposed HMFO

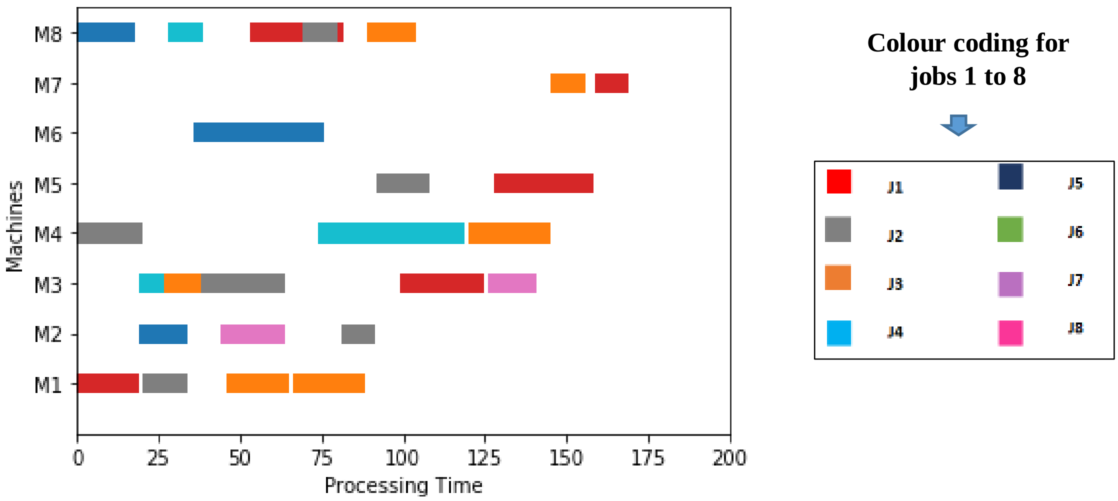


Figure 6.25 Gantt charts for the scenario 3 for the proposed HMFO

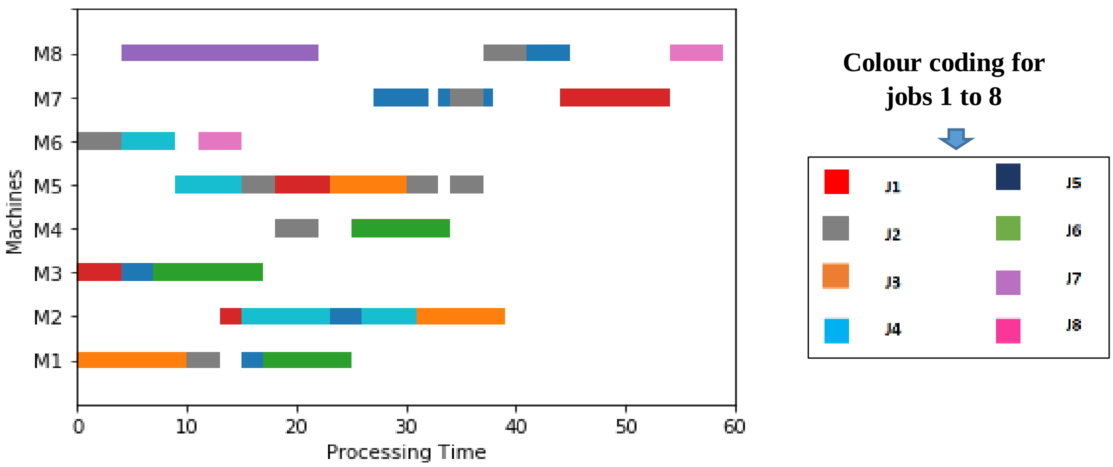


Figure 6.26 Gantt charts for the scenario 4 for the proposed HMFO

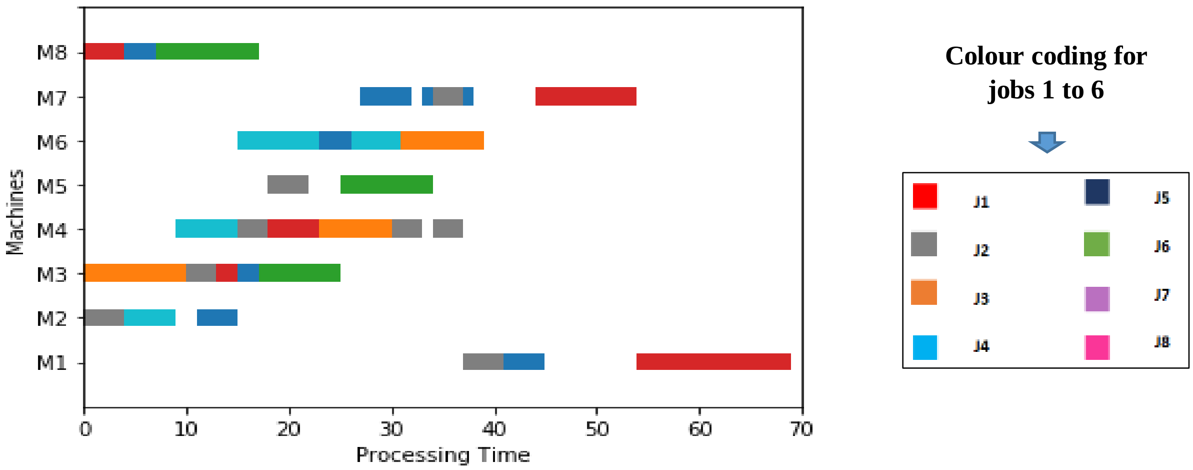


Figure 6.27 Gantt charts for the scenario 5 for the proposed HMFO

Apart from the comparison of both proposed HMFO 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. 6.37. For various scenarios, the machine utilization for machines is different i.e., from Fig. 6.37 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. 6.36., and comparing all the scenarios 1 to 10, it is possible to realize that the proposed HMFO gives comparatively gives better Pareto- optimal results when compared to the NSGA III algorithm.

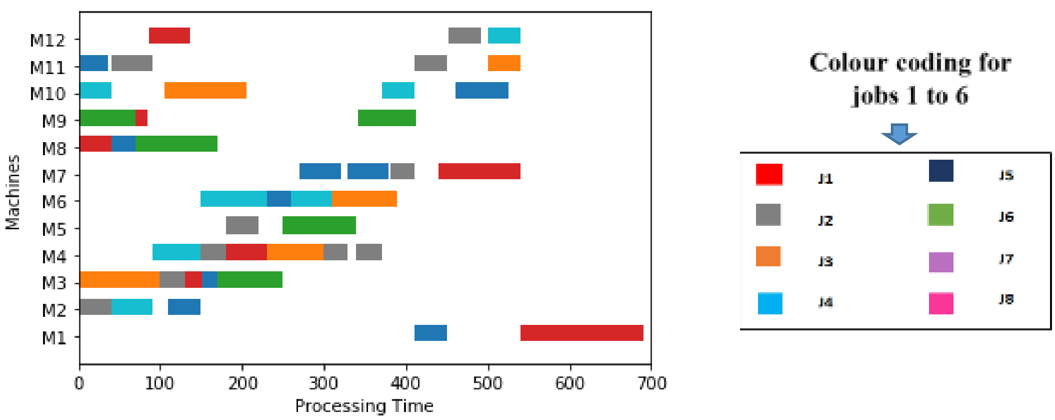


Figure 6.28 Gantt charts for the scenario 6 for the proposed HMFO

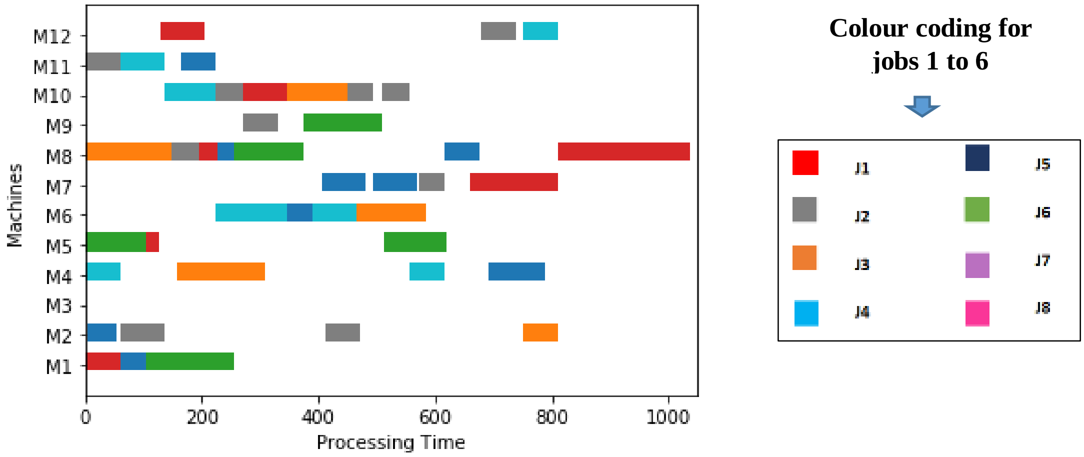


Figure 6.29 Gantt charts for the scenarios 7 for the proposed HMFO

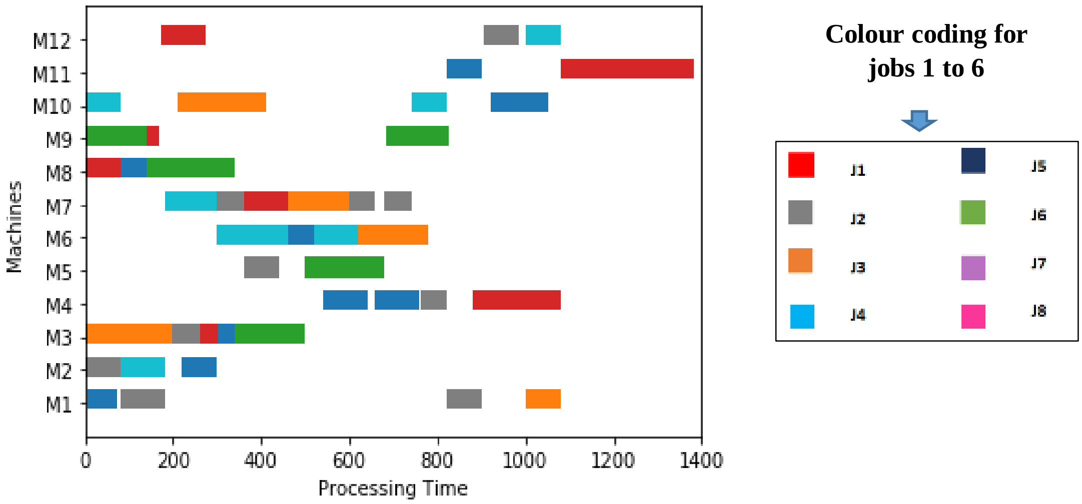


Figure 6.30 Gantt charts for the scenarios 8 for the proposed HMFO

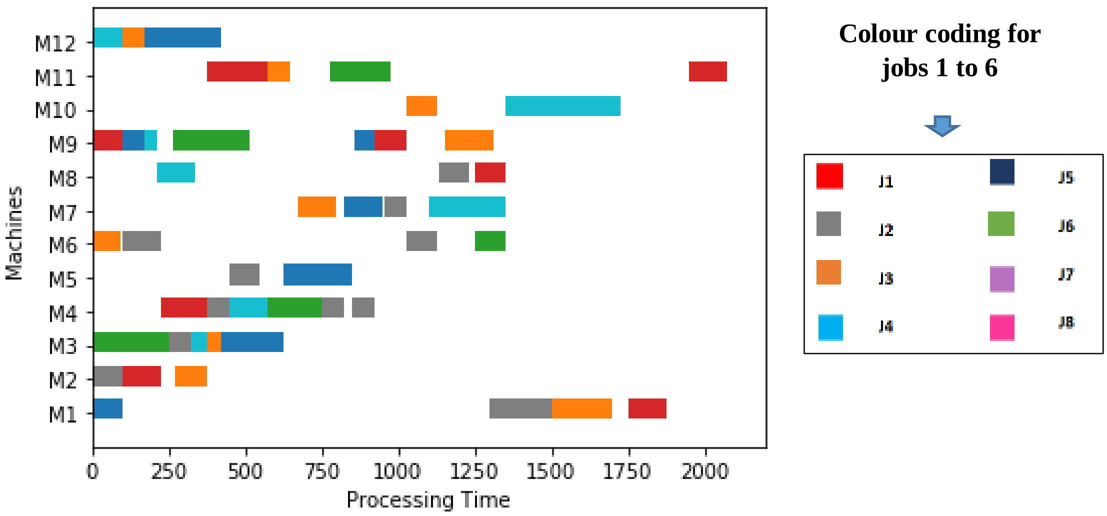


Figure 6.31 Gantt charts for the scenarios 9 for the proposed HMFO

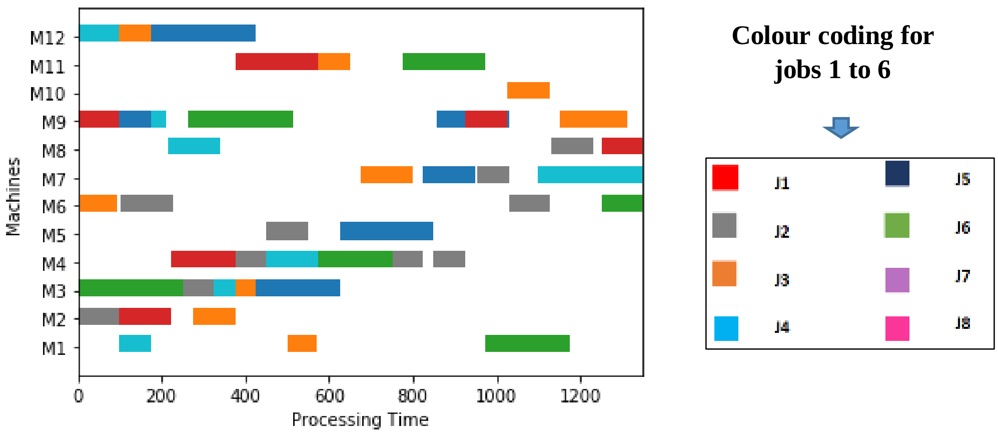


Figure 6.32 Gantt charts for the scenario10 for the proposed HMFO

In addition, to comparing the performance of both the algorithms i.e., HMFO and NSGA III a comparative study of the machine utilization rate of different machines for all the ten instances is illustrated in Fig. 6.23, Fig. 6.24, Fig. 6.25, Fig. 6.26, Fig. 6.27, Fig. 6.28, Fig. 6.29, Fig. 6.30, Fig. 6.31, Fig. 6.32 respectively.

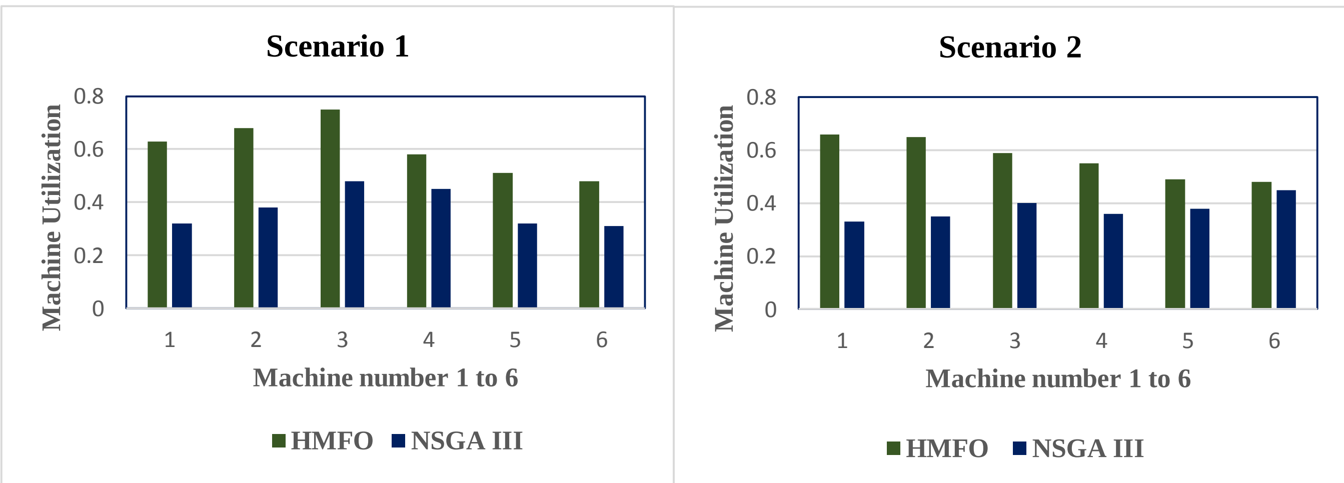
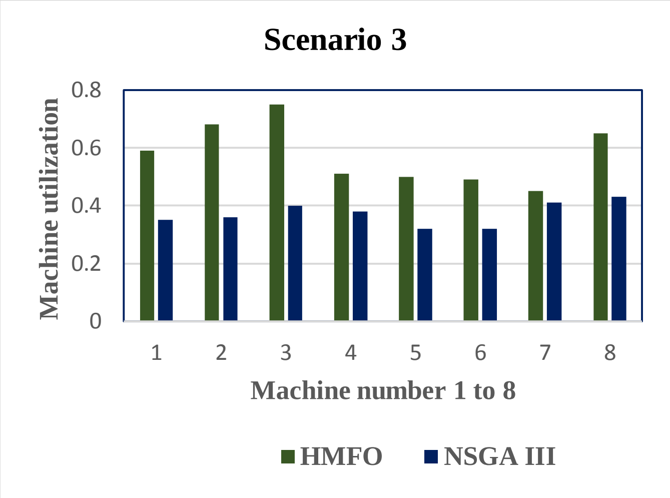


Figure 6.33 Comparison of Machine Utilization for the scenarios 1 (Left) to 2 (Right) for HMFO and NSGA III



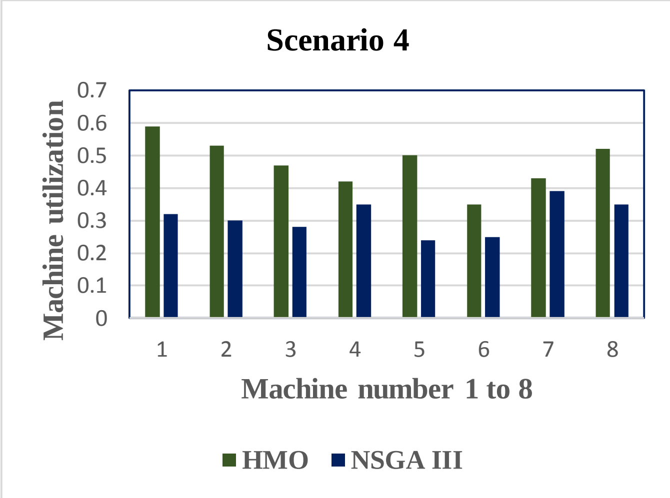


Figure 6.34 Machine Utilization values for the scenarios 3 (Left) and 4 (Right) for HMFO and NSGA III.

It can be inferred from Fig.6.33 to Fig. 6.35 all scenarios 1 to 5 machine utilization rates are far better for the results that are obtained with HMFO when compared with results that are obtained with NSGA III. From Fig. 4.19 for scenario 7 in case of HMFO, machine 4 has the maximum utilization rate and machine 11 has the minimum utilization rate and in case of NSGA-III. Similarly, the scenario 10 also the machine 4 has the highest and machine 11 has the least utilization rate.

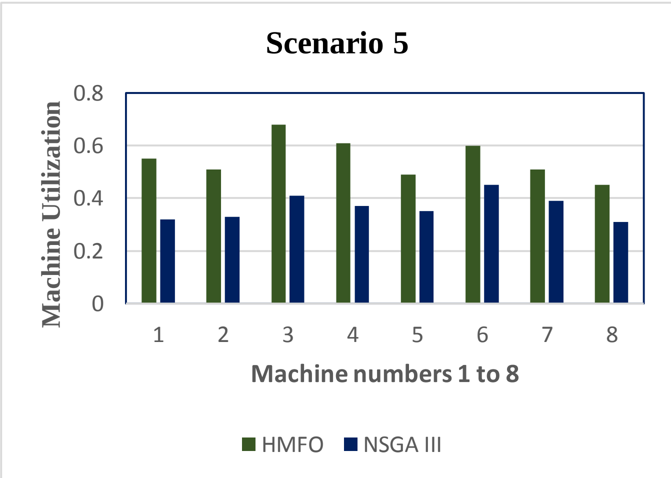
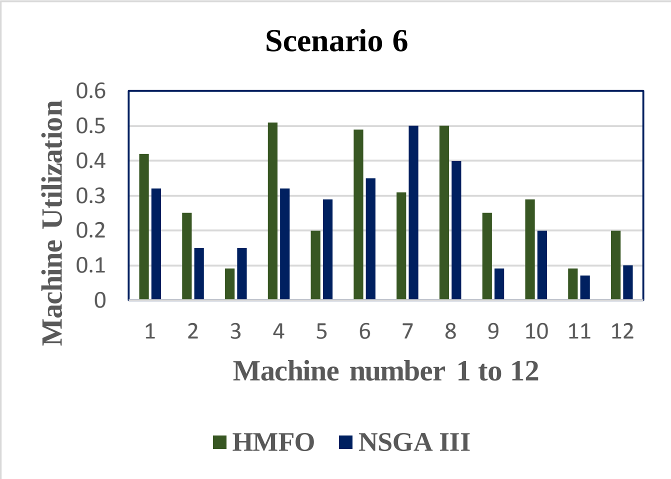


Figure 6.35 Comparison of Machine Utilization of all machines in scenarios 5 for HMFO and NSGA III.



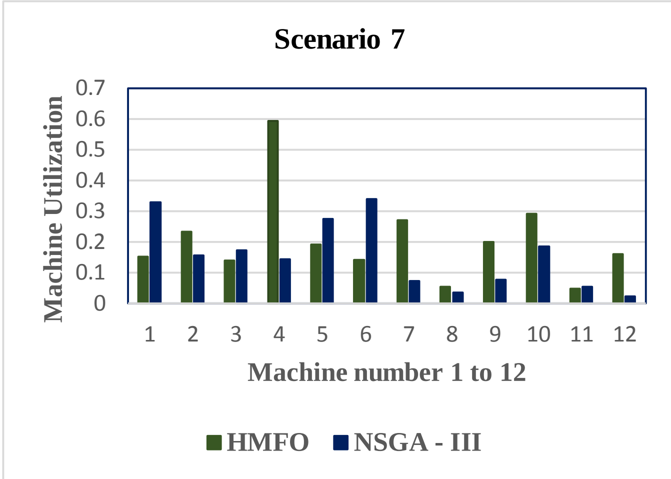
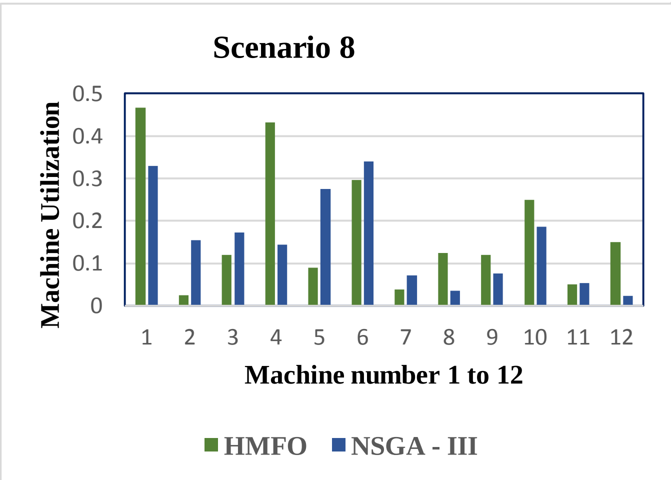


Figure 6.36 Machine Utilization for the scenarios 6 and 7 for HMFO and NSGA III.



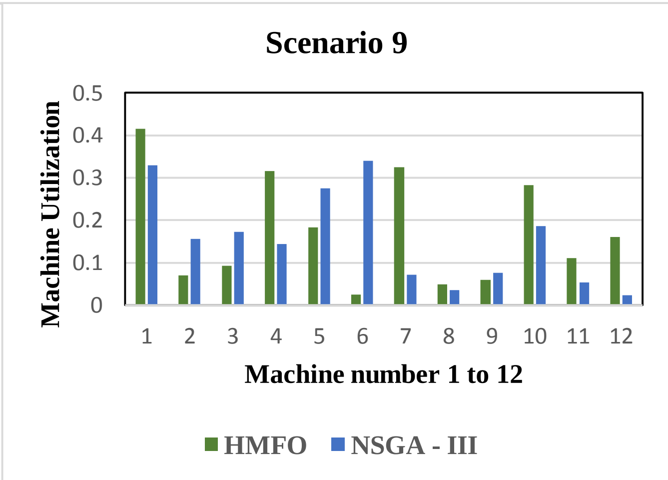


Figure 6.37 Machine Utilization for the scenarios 8 and 9 for HMFO and NSGA III.

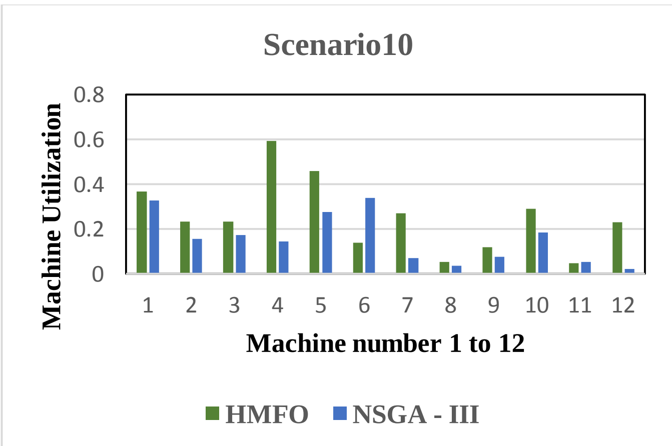


Figure 6.38 Machine Utilization for the scenarios 10 for HMFO and NSGA III.

The Energy Consumption data is shown with a bar chart shown in Fig. 6.39 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 HMFO and NSGA III algorithms the pattern is similar. But the proposed HMFO gives lower values of energy consumption indicating better performance over the NSGA III algorithm.

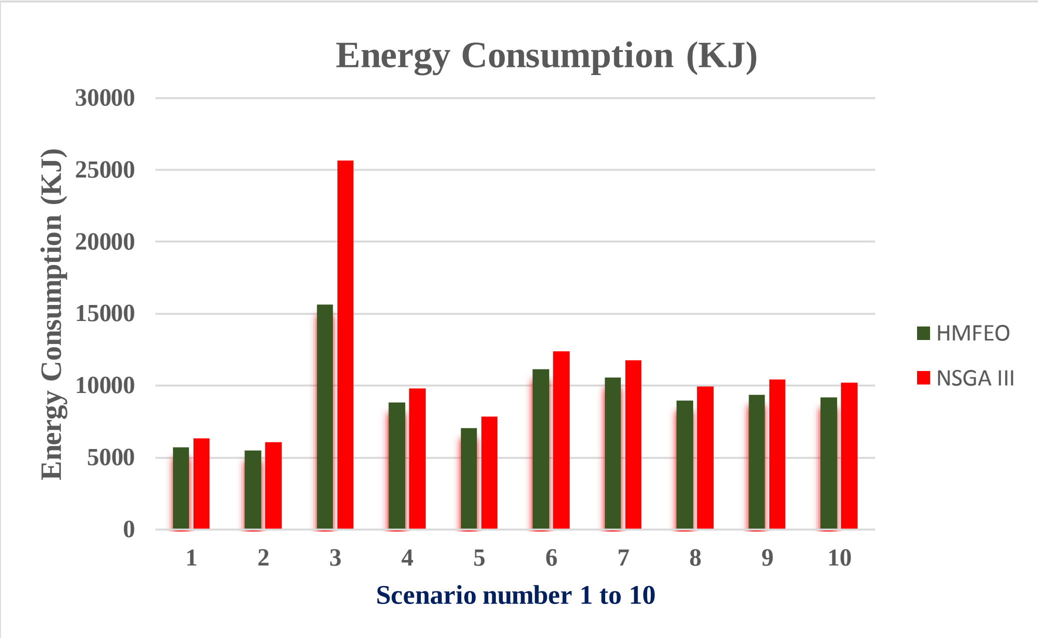


Figure 6.39. Energy consumption for all the scenario for the proposed HMFO and NSGA III

6.8. Various Performance indicators for validity of proposed hybrid moth flame optimization with NSGA-III.

Several performance indicators (PI) were suggested by [92-95] and these indicators compares the performance of the multi/ many objective algorithms. Mostly used Performance Measure out of all is the hyper volume (Deng et al., 2019) . Hyper volume (HV) is the volume surrounded by the dominated Pareto front approximation 'K' from a reference point x £ X^p, such that b £ K, K<x. The HV is given by Equation (14). Here, η_P represents P dimensional lebesgue measure.

$$HV(K,\ x) = \eta_P\left(\bigcup_{B \in K} [B, x]\right) \tag{6.14}$$

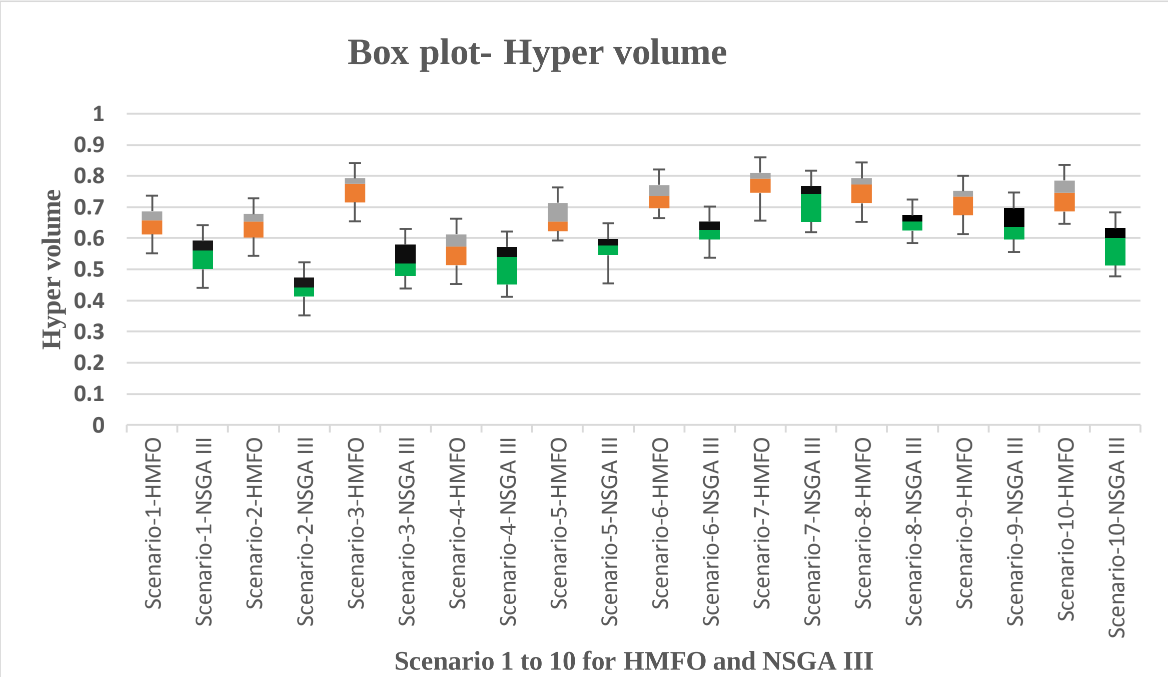
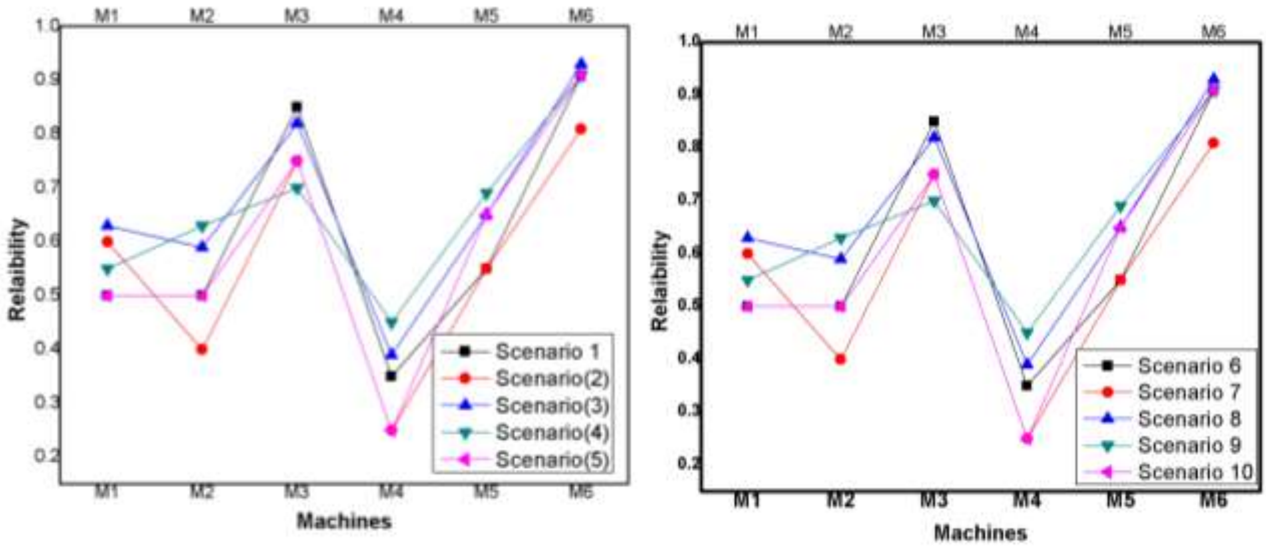


Figure 6.40 Box plot indicating hypervolume values for all the ten scenarios of HMFO and NSGA-III.



(a) (b)

Figure 6.41 (a) Reliability graphs for the scenario1 to scenario 5, **(b)** Reliability graphs for the –scenario 6 to scenario 10 for the proposed MFEO

The reliability of various machines for scenario 1 to scenario 5 is plotted in Fig. 6.37. (a) and for comparison purposes only the first six machines of all the scenarios taken into consideration. From the Fig. 6.37. (a) it can be deduced for scenario 1 to 5 the reliability for the machine 2 and machine 5 is less when compared to the other machines. The reliability values are varying for different machines but for all the scenarios, the trend is the same. From this Fig. 6.37. (a) more concentration on the improvement in processing the machines that are concerned with machine 2 and machine 5 is needed for further improvement in the results of objective functions. Fig. 6.37. (b) represents the reliability of various machines for scenario 6 to scenario 10. The reliability values for machines 2 and 5 are lower when compared with other machines for scenarios 6 to scenarios 10. Reliability for machine 3 is higher for all the scenarios (i.e. 1 to 10) which indicates the concern operations are happening in an uninterrupted manner.

The HV values for all ten scenarios are plotted with the help of a box plate for the proposed HMFO 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. 6.37. The higher the HV indicates, the better is the performance. Fig. 6.37 depicts that the proposed HMFO is better than the NSGA III algorithm. From the Fig. 6.37 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.

Table 6.10 Performance Indicators for all Scenario 1 to 5 for both HMFO and NSGA -III

Indicator	**Algorithm**	**Scenario**				
		I	II	III	IV	V
χ	HMFO	**9.0**	**11.7**	**9.7**	**10.0**	**9.1**
	NSGA III	8.1	10.5	8.9	10.0	8.4
ψ	HMFO	**8.7**	**10.6**	**9.5**	**10.0**	**8.5**
	NSGA III	7.5	9.5	8.1	9.0	8.1
χ/ψ	HMFO	**0.9666**	**0.9900**	**0.9793**	**1.0000**	**0.9340**
	NSGA III	0.8750	0.9047	0.9101	0.9	0.9642

ω	HMFO	**0.5485**	**0.5124**	**0.6245**	**0.5245**	**0.5652**
	NSGA III	0.4375	0.4987	0.3985	0.5196	0.4841
κ	HMFO	**0.0700**	**0.0156**	**0.0320**	**0.0012**	**0.0300**
	NSGA III	0.1300	0.0894	0.0116	0.1333	0.0412
∂	HMFO	**0.4256**	**0.0042**	**0.7378**	**0.0023**	**0.0068**
	NSGA III	11.321	8.666	7.345	9.5321	9.6631
£	HMFO	0.3176	**0.4963**	**0.4814**	**0.4785**	**0.4258**
	NSGA III	**0.4569**	0.5666	0.6325	0.6841	0.6124
CPU TIME(s)	HMFO	**137.8**	**140.3**	**142**	**139.1**	**141.7**
	NSGA III	246	216	263.4	243.6	256.3

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 6.10 and for scenarios 6 to 10 the PM values are represented in Table 6.11. 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.

$$\omega = \frac{\left| C(\underset{j}{\cup} N_j) \setminus C(\underset{j \neq l}{\cup} N_j) \right|}{C(\underset{j}{\cup} N_j)} \qquad (6.15)$$

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

Table 6.11 Performance Indicators for all Scenario 6 to 10 for both HMFO and NSGA -III

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

κ (p,q) in Equation (6.16) 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.

$$\kappa(p,q) = \frac{|q \in Q, \exists p \in P : p > q|}{|Q|} \qquad (6.16)$$

Lesser λ is necessary, and the values which are very nearer to zero indicate the highly distributed uniformly over the Pareto front. Equation (6.17) 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 (∂) measure in Equation (6.17) 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 between two consecutive points average Euclidian length $\bar{G}$ over all the available non-Dominated solutions.

$$\partial = \frac{G_f + G_l + \sum_{i=1}^{J-1} \left| G_i - \bar{G} \right|}{G_f + G_l + (J-1)\bar{G}} \qquad (6.17)$$

£ 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 6.10 and Table 6.11. The CPU time for the proposed algorithm is less than the NSGA III indicating the superiority of the algorithm.

6.9. 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 HMFO 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.

6.10 Conclusions

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 work 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 HMFO and tuned the algorithm to fit the intended problem objectives. Initially the proposed HMFO algorithm is tested for various small sized test instances with the CPLEX solver. The results showed the superiority of the proposed HMFO over CPLEX solver. Later medium sized problem scenarios were tested and compared the performance of HMFO with NSGA-III. The results demonstrate that the use of HMFO 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 HMFO. Thus, the results obtained showcase the effectiveness of the approach mentioned in this research. In this work, a fewer number of enterprises in a DMS and application of public permission-less blockchain based smart contract to find the potential enterprises in the DMS has been considered where the permsionless blockchain can be considered as a scope for improvement. 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.

Chapter 7

Conclusions and scope of the future work

7.1 Introduction

This chapter focuses on the consolidation of research findings in the area of networked manufacturing systems' in the context of integrated process planning and scheduling. Here, the functional differences, benefits, and drawbacks of traditional and networked manufacturing systems has briefly mentioned. The flexible process planning and scheduling has been mapped and linked with industrial inputs relating to data and customer requirements. The most recent algorithms that draw inspiration from nature, text mining techniques based on machine learning, and blockchain based methods has been used to tackle the problems' objectives and restrictions. The viability of the suggested approach is confirmed using several difficult circumstances and an illustrative case. The network-based manufacturing system's sustainability was calculated using sustainable performance criteria.

7.2 Application Domain

- Supporting remote and collaborative operations is necessary due to consumer customization. Data mining is a useful method for finding resources and meeting the needs of a networked manufacturing environment.

- Using an efficient strategy in the context of networked production to minimize the manufacturing time and maximize machine utilization, as well as to minimize energy consumption and increase reliability. We have developed various models in this study to replicate the real manufacturing settings.

- As part of this research, we created a multi-objective formulation with a variety of goals, including minimizing makespan. We also created goals that improve sustainability goals, like maximizing machine utilization and minimizing energy use, with machine and operation sequence as its constraints.

- We have developed different multi/ many objective evolutionary algorithms to solve the multi-objective problems proposed in this research and find out the Pareto optimal solutions.

- In this research, apart from increase in the security of information, we have proposed a blockchain based approach to share the resources that help to integrate the manufacturing functions in a distributed manner, and its fundamental framework and functions are presented.

7.3 Contribution to thesis

Several contributions made in this thesis are mentioned below.

- The developed data mining assisted process planning and scheduling serves the researches to solve more complicated flexible process planning and scheduling problems.

- There is a greater need of effective MOEA, the proposed HMFO is one such algorithm to satisfy the needs of MOPs.

- Proposed Block chain based smart contract can help to overcome the security and transparency issues for distributed environment based research.

7.4 Scope for Future Research

1. Process planning and scheduling complexity in networked manufacturing: Multi-objective formulation and solutions.

2. Resource scalability in networked manufacturing system: A machine learning based approach can be a scope for improvement.

3. A case of network based manufacturing environment to explore the key elements in the system with an integrated blockchain and mobile agent systems.

4. Evaluation of various performance measures of with integration of various blockchain techniques with cyber physical system on networked manufacturing system.

Publications from the Research Work

International Journals:

1. **Ramakurthi, V.B**.; Manupati, V.K.; Machado, J.; Varela, L. A Hybrid Multi-Objective Evolutionary Algorithm-Based Semantic Foundation for Sustainable Distributed Manufacturing Systems. *Appl. Sci.* **2021**, *11*, 6314. https://doi.org/10.3390/app11146314 .SCIE. **I.F -(2.679)**

2. **Ramakurthi, V. B**., Manupati, V. K., Varela, L., & Machado, J. (2020, September). Energy Efficient Network Manufacturing System Using Controlled Elitist Non-dominated Sorting Genetic Algorithm. *International Journal of Mechatronics and Applied Mechanics, 2020, Issue 7* **Scopus** DOI: 10.1007/978-3-030-53973-3_21.

3. **Ramakurthi, V. B**., Manupati, V. K., Suresh babu E, Varela, L., & Machado, J An innovative approach for resource sharing and scheduling in a sustainable distributed manufacturing system. *Advanced Engineering Informatics,* SCIE **I.F -(7.862)**

International Conference:

1. **Ramakurthi, V**., Manupati, V., Varela, M. L. R., & Putnik, G. (2021, June). A Novel Integrated Framework Approach for TEBC Technologies in Distributed Manufacturing Systems: A Systematic Review and Opportunities. In *International Conference Innovation in Engineering* (pp. 101-112). **Springer, Cham.** DOI: 10.1007/978-3-030-79165-0_10

Book Chapter:

1. **Ramakurthi V.B**, Vijaya Kumar Manupati, Nikhil Wakode, M.L.R. Varela (2020). Ensuring Sustainability in Industry 4.0: Implementation Framework. In Sustainable Manufacturing for Industry 4.0 (pp. 215-252). CRC Press published in book Sustainable Manufacturing for Industry 4.0: An Augmented Approach (1st ed.). **CRC Press**. https://doi.org/10.1201/9780429466298,

References

1. Mezgár, I., & Kovács, G. L. (1997). A network-based manufacturing management system. *IFAC Proceedings Volumes*, *30*(1), 169-174.

2. Maravelias, C. T., & Sung, C. (2009). Integration of production planning and scheduling: Overview, challenges and opportunities. *Computers & Chemical Engineering*, *33*(12), 1919-1930.

3. Yan, J. (2000). Digitalization and networked manufacturing. *Ind Eng Manage*, *1*, 8-11.

4. Tranfield, D., Denyer, D., & Smart, P. (2003). Towards a methodology for developing evidence-informed management knowledge by means of systematic review. *British journal of management*, *14*(3), 207-222.

5. Denyer, D., & Tranfield, D. (2009). Producing a systematic review.

6. Tan, W. L., Wong, T. L., Wong, M. C., & Lang, N. P. (2012). A systematic review of post-extractional alveolar hard and soft tissue dimensional changes in humans. *Clinical oral implants research*, *23*, 1-21.

7. Strozzi, F., & Colicchia, C. (2012). Literature review on complex network methods applied to measure robustness in supply chain design.

8. Choudhary, Alok Kumar, P. I. Oluikpe, Jenny A. Harding, and Patricia M. Carrillo. (2009): "The needs and benefits of Text Mining applications on Post-Project Reviews." *Computers in Industry* 60, no. 9 728-740. https://doi.org/10.1016/j.compind.2009.05.006

9. Li, Nan, and Desheng Dash Wu. (2010) "Using text mining and sentiment analysis for online forums hotspot detection and forecast." *Decision support systems* 48, no. 2: 354-368. https://doi.org/10.1016/j.dss.2009.09.003

10. Tursi, Angela, Hervé Panetto, Gérard Morel, and Michele Dassisti (2009). Ontological approach for products-centric information system interoperability in networked manufacturing enterprises. *Annual Reviews in Control* 33, no. 2, 238-245.. 10.1016/j.arcontrol.2009.05.003ff. ffhal-00361394v2f

11. Yazdizadeh, Peyman, and Farhad Ameri (2015). A text mining technique for manufacturing supplier classification." In *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, V01BT02A036-

V01BT02A036, American Society of Mechanical Engineers. https://doi.org/10.1115/DETC2015-46694

12. Morariu, C., Morariu, O., Răileanu, S., & Borangiu, T. (2020). Machine learning for predictive scheduling and resource allocation in large scale manufacturing systems. *Computers in Industry*, *120*, 103244. https://doi.org/10.1016/j.compind.2020.103244

13. Park, Kijung, and Gül E. Okudan Kremer (2017). Text mining-based categorization and user perspective analysis of environmental sustainability indicators for manufacturing and service systems. *Ecological indicators* 72, 803-820. https://doi.org/10.1016/j.ecolind.2016.08.027

14. Akhtar, M. D., Vijaya Kumar Manupati, Maria Leonilde R. Varela, Goran D. Putnik, Ana Maria Madureira, and Ajith Abraham. (2017) "Manufacturing Services Classification in a Decentralized Supply Chain Using Text Mining." In *International Conference on Health Information Science*, pp. 186-193. Springer, Cham,. https://doi.org/10.1007/978-3-319-76351-4_19

15. Bianchi, Gianpiero, Renato Bruni, and Francesco Scalfati. (2018). "Identifying e-Commerce in Enterprises by means of Text Mining and Classification algorithms." *Mathematical Problems in Engineering* https://doi.org/10.1155/2018/7231920

16. Shao, X., Li, X., Gao, L., & Zhang, C. (2009). Integration of process planning and scheduling—a modified genetic algorithm-based approach. *Computers & Operations Research*, *36*(6), 2082-2096. https://doi.org/10.1016/j.cor.2008.07.006

17. Chryssolouris, G., S. Chan, and W. Cobb. (1984) "Decision making on the factory floor: an integrated approach to process planning and scheduling." *Robotics and Computer-Integrated Manufacturing* 1, no. 3-4: 315-319. https://doi.org/10.1016/S0007-8506(07)61801-0

18. Chryssolouris, G., S. Chan, and N. P. Suh. (1985) "An integrated approach to process planning and scheduling." *CIRP Annals* 34, no. 1: 413-417. https://doi.org/10.1016/S0007-8506(07)61801-0

19. Montreuil, Benoit, Jean-Marc Frayret, and Sophie D'Amours. (2000) "A strategic framework for networked manufacturing." *Computers in industry* 42, no. 2-3: 299-317. https://doi.org/10.1016/S0166-3615(99)00078-0

20. Wang, Y. H., C. W. Yin, and Y. Zhang. (2003)"A multi-agent and distributed ruler based approach to production scheduling of agile manufacturing systems." *International Journal of Computer Integrated Manufacturing* 16, no. 2: 81-92. https://doi.org/10.1080/713804987

21. Chen, C.-L. A Novel Disruptive Innovation-Like Algorithm for Single-Machine Scheduling with Sequence-Dependent Family Setup Times. *Appl. Sci.* **2021**, *11*, 986. https://doi.org/10.3390/app11030986

22. Tursi, Angela, Hervé Panetto, Gérard Morel, and Michele Dassisti. (2009)"Ontological approach for products-centric information system interoperability in networked manufacturing enterprises." *Annual Reviews in Control* 33, no. 2: 238-245. https://doi.org/10.1016/j.arcontrol.2009.05.003

23 Dehghani, M.; Montazeri, Z.; Dehghani, A.; Malik, O.P.; Morales-Menendez, R.; Dhiman, G.; Nouri, N.; Ehsanifar, A.; Guerrero, J.M.; Ramirez-Mendoza, R.A. Binary Spring Search Algorithm for Solving Various Optimization Problems. *Appl. Sci.* **2021**, *11*, 1286. https://doi.org/10.3390/app11031286

24. Manupati, Vijaya Kumar, Jitesh J. Thakkar, Kuan Yew Wong, and Manoj Kumar Tiwari. (2013) "Near optimal process plan selection for multiple jobs in networked based manufacturing using multi-objective evolutionary algorithms." *Computers & Industrial Engineering* 66, no. 1: 63-76. https://doi.org/10.1016/j.cie.2013.06.004

25. Ehm, Jens, Michael Freitag, and Enzo M. Frazzon. (2016) "A heuristic optimisation approach for the scheduling of integrated manufacturing and distribution systems." *Procedia CIRP* 57 :357-361. http://creativecommons.org/licenses/by-nc-nd/4.0/)

26. Al-Shayea, A., Fararah, E., Nasr, E. A., & Mahmoud, H. A. (2020). “Model for Integrating Production Scheduling and Maintenance Planning of Flow Shop Production System”. *IEEE Access*, *8*, 208826-208835. DOI: 10.1109/ACCESS.2020.3038719

27. Manupati, V. K., Sujay Deo, N. Cheikhrouhou, and M. K. Tiwari. (2012) "Optimal process plan selection in networked based manufacturing using game-theoretic

approach." *International Journal of Production Research* 50, no. 18: 5239-5258. https://doi.org/10.1080/00207543.2012.682181

28. Petrović, Milica, Marko Mitić, Najdan Vuković, and Zoran Miljković. (2016) "Chaotic particle swarm optimization algorithm for flexible process planning." *The International Journal of Advanced Manufacturing Technology* 85, no. 9-12: 2535-2555. DOI 10.1007/s00170-015-7991-4

29. Mohammadi, S., Al-e-Hashem, S. M., & Rekik, Y. (2020). "An integrated production scheduling and delivery route planning with multi-purpose machines: A case study from a furniture manufacturing company". *International Journal of Production Economics, 219*, 347-359. https://doi.org/10.1016/j.ijpe.2019.05.017

30. Barzanji, Ramin, Bahman Naderi, and Mehmet A. Begen. (2019) "Decomposition algorithms for the integrated process planning and scheduling problem." *Omega* . DOI: 10.1016/j.omega.2019.01.003

31. Ertogral, Kadir, and Fatma Sedanur Öztürk (2019). An integrated production scheduling and workforce capacity planning model for the maintenance and repair operations in airline industry. *Computers & Industrial Engineering* 127, 832-840. https://doi.org/10.1016/j.cie.2018.11.022

32. Li, Xinyu, Liang Gao, Wenwen Wang, Cuiyu Wang, and Long Wen (2019). Particle swarm optimization hybridized with genetic algorithm for uncertain integrated process planning and scheduling with interval processing time. *Computers & Industrial Engineering*. https://doi.org/10.1016/j.cie.2019.04.028

33. Cao, Y., Zhang, H., Li, W., Zhou, M., Zhang, Y., & Chaovalitwongse, W. A. (2018). Comprehensive learning particle swarm optimization algorithm with local search for multimodal functions. IEEE Transactions on Evolutionary Computation, 23(4), 718-731.

doi: 10.1109/TEVC.2018.2885075.

34. Shaw, Michael J. (1987). A distributed scheduling method for computer integrated manufacturing: the use of local area networks in cellular systems. *International Journal of Production Research* 25, no. 9, 1285-1303. http://pascal-francis.inist.fr/vibad/index.php?action=getRecordDetail&idt=7738721

35. Maturana, Francisco P., and Douglas H. Norrie (1995). A generic mediator for multi-agent coordination in a distributed manufacturing system. In *1995 IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century*, vol. 1, 952-957. DOI: 10.1109/ICSMC.1995.537890

36._Sohrabi, M. K., & Azgomi, H. (2020). A survey on the combined use of optimization methods and game theory. *Archives of Computational Methods in Engineering*, *27*(1), 59-80. DOI :https://doi.org/10.1007/s11831-018-9300-5

37. Jia, H. Z., Jerry YH Fuh, Andrew YC Nee, and Y. F. Zhang (2007). Integration of genetic algorithm and Gantt chart for job shop scheduling in distributed manufacturing systems. *Computers & Industrial Engineering 53*, no. 2, 313-320. https://doi.org/10.1016/j.cie.2007.06.024

38. Wang, Y. H., C. W. Yin, and Y. Zhang (2003). A multi-agent and distributed ruler based approach to production scheduling of agile manufacturing systems. *International Journal of Computer Integrated Manufacturing* 16, no. 2, 81-92. https://doi.org/10.1080/713804987.

39. Jian Gao, Yida Xu, Julian Barreiro-Gomez, Massa Ndong, Michalis Smyrnakis and Hamidou Tembine (September 5th 2018). Distributionally Robust Optimization, Optimization Algorithms - Examples, Jan Valdman, IntechOpen, DOI: 10.5772/intechopen.76686.

40. Liu, Bo, Keyao Wang, and Ran Zhang (2016). Variable neighborhood based memetic algorithm for distributed assembly permutation flowshop. In *2016 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1682-1686. IEEE. DOI: 10.1109/CEC.2016.7743990

41. Kesen, Saadettin Erhan, Sanchoy K. Das, and Zülal Güngör (2010). A genetic algorithm based heuristic for scheduling of virtual manufacturing cells (VMCs). *Computers & Operations Research* 37, no. 6, 1148-1156. https://doi.org/10.1016/j.cor.2009.10.006

42. Sortrakul, N., Nachtmann, H. L., & Cassady, C. R. (2005). Genetic algorithms for integrated preventive maintenance planning and production scheduling for a single machine. *Computers in Industry*, *56*(2), 161-168. https://doi.org/10.1016/j.compind.2004.06.005

43. Li, W. D., and Chris A. McMahon. (2007) "A simulated annealing-based optimization approach for integrated process planning and scheduling." *International Journal of Computer Integrated Manufacturing* 20, no. 1: 80-95. https://doi.org/10.1080/09511920600667366

44. Tiwari, Anurag, Pei-Chann Chang, M. K. Tiwari, and Rishabh Kandhway (2016). A Hybrid Territory Defined evolutionary algorithm approach for closed loop green supply chain network design. *Computers & Industrial Engineering* 99, 432-447. https://doi.org/10.1016/j.cie.2016.05.018

45. Li, Yongfeng, Fei Tao, Ying Cheng, Xianzhi Zhang, and A. Y. C. Nee (2017). Complex networks in advanced manufacturing systems. *Journal of Manufacturing Systems* 43, 409-421. https://doi.org/10.1016/j.jmsy.2016.12.001

46. Tang, D., Dai, M., Salido, M. A., & Giret, A. (2016). Energy-efficient dynamic scheduling for a flexible flow shop using an improved particle swarm optimization. *Computers in Industry*, *81*, 82-95. https://doi.org/10.1016/j.compind.2015.10.001

47. Hou, Y., Wu, N., Zhou, M., & Li, Z. (2015). Pareto-optimization for scheduling of crude oil operations in refinery via genetic algorithm. *Ieee transactions on systems, man, and cybernetics: systems*, *47*(3), 517-530. doi: 10.1109/TSMC.2015.2507161.

48. Maaranen, H., Miettinen, K., & Penttinen, A. (2007). On initial populations of a genetic algorithm for continuous optimization problems. *Journal of Global Optimization*, *37*(3), 405-436. DOI 10.1007/s10898-006-9056-6.

49. Lin, Lin, Xin-Chang Hao, Mitsuo Gen, and Jung-Bok Jo (2012). Network modeling and evolutionary optimization for scheduling in manufacturing. *Journal of Intelligent Manufacturing* 23, no. 6, 2237-2253. DOI 10.1007/s10845-011-0569-6

50. Piroozfard, H., Wong, K. Y., & Hassan, A. (2016). A hybrid genetic algorithm with a knowledge-based operator for solving the job shop scheduling problems. *Journal of Optimization*, *2016*. https://doi.org/10.1155/2016/7319036

51. Klement, N.; Abdeljaouad, M.A.; Porto, L.; Silva, C. Lot-Sizing and Scheduling for the Plastic Injection Molding Industry—A Hybrid Optimization Approach. *Appl. Sci.* **2021**, *11*, 1202. https://doi.org/10.3390/app11031202

52. Tursi, A., Panetto, H., Morel, G., & Dassisti, M. (2007). Ontology-based products information interoperability in networked manufacturing enterprises. IFAC Proceedings Volumes, 40(19), 85-93. https://doi.org/10.3182/20071002-MX-4-3906.00015
53. Manupati, V. K., Putnik, G. D., Tiwari, M. K., Ávila, P., & Cruz-Cunha, M. M. (2016). Integration of process planning and scheduling using mobile-agent based approach in a networked manufacturing environment. Computers & Industrial Engineering, 94, 63-73. https://doi.org/10.1016/j.cie.2016.01.017
54. Guo, S., & Dong, M. (2020). Order matching mechanism of the production intermediation internet platform between retailers and manufacturers. The International Journal of Advanced Manufacturing Technology, 1-14. https://doi.org/10.1007/s00170-020-06175-z
55. Abdullah, N., Hakansson, A., & Moradian, E. (2017). 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). IEEE. DOI: 10.1109/ICUFN.2017.7993927
56. Lu, Y. (2019). The blockchain: State-of-the-art and research challenges. Journal of Industrial Information Integration, 15, 80-90. https://doi.org/10.1016/j.jii.2019.04.002
57. Sharma, P. K., Kumar, N., & Park, J. H. (2018). Blockchain-based distributed framework for automotive industry in a smart city. IEEE Transactions on Industrial Informatics, 15(7), 4197-4205. DOI: 10.1109/TII.2018.2887101
58. Li, Z., Barenji, A. V., & Huang, G. Q. (2018). Toward a blockchain cloud manufacturing system as a peer to peer distributed network platform. Robotics and computer-integrated manufacturing, 54, 133-144. https://doi.org/10.1016/j.rcim.2018.05.011
59. Risius, M., & Spohrer, K. (2017). A blockchain research framework. Business & Information Systems Engineering, 59(6), 385-409. https://doi.org/10.1007/s12599-017-0506-0
60. Yang, R., Yu, F. R., Si, P., Yang, Z., & Zhang, Y. (2019). Integrated blockchain and edge computing systems: A survey, some research issues and challenges. IEEE Communications Surveys & Tutorials, 21(2), 1508-1532. doi: 10.1109 /COMST. 2019.2894727.
61. Viriyasitavat, W., Da Xu, L., Bi, Z., & Sapsomboon, A. (2018). Blockchain-based business process management (BPM) framework for service composition in industry 4.0. Journal of Intelligent Manufacturing, 1-12. https://doi.org/10.1007/s10845-018-1422-y
62. Udokwu, C., Kormiltsyn, A., Thangalimodzi, K., & Norta, A. (2018). An exploration of blockchain enabled smart-contracts application in the enterprise. Technical Report, 1-29.

DOI: 10.13140/RG.2.2.36464.97287

63. Johng, H., Kim, D., Hill, T., & Chung, L. (2018, July). 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. DOI: 10.1145/3284764.3284768
64. Chung, K., Yoo, H., Choe, D., & Jung, H. (2019). Blockchain network based topic mining process for cognitive manufacturing. Wireless Personal Communications, 105(2), 583-597. https://doi.org/10.1007/s11277-018-5979-8
65. Supranee, S., & Rotchanakitumnuai, S. (2017). 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). https://aisel.aisnet.org/iceb2017/30
66. Angrish, A., Craver, B., Hasan, M., & Starly, B. (2018). A case study for Blockchain in manufacturing:“FabRec”: A prototype for peer-to-peer network of manufacturing nodes. Procedia Manufacturing, 26, 1180-1192. https://doi.org/ 10.1016/j.promfg. 2018.07.154
67. Westerkamp, M., Victor, F., & Küpper, A. (2020). Tracing manufacturing processes using blockchain-based token compositions. Digital Communications and Networks, 6(2), 167-176. https://doi.org/10.1016/j.dcan.2019.01.007
68. Zafar, M. A., Sher, F., Janjua, M. U., & Baset, S. (2018, December). 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). DOI: 10.1145/3284764.3284768
69. Bogner, A., Chanson, M., & Meeuw, A. (2016, November). A decentralised sharing app running a smart contract on the ethereum blockchain. In Proceedings of the 6th International Conference on the Internet of Things (pp. 177-178). DOI: http://dx.doi.org/10.1145/2991561.2998465
70. Hasan, H. R., Salah, K., Jayaraman, R., Ahmad, R. W., Yaqoob, I., & Omar, M. (2020). Blockchain-based solution for the traceability of spare parts in manufacturing. IEEE Access, 8, 100308-100322. DOI: 10.1109/access.2020.2998159
71. Lohmer, J. (2019). Applicability of blockchain technology in scheduling resources within distributed manufacturing. In Logistics Management (pp. 89-103). Springer, Cham. : https://doi.org/10.1007/978-3-030-29821-0_7

72. Menezes, G. C., Mateus, G. R., & Ravetti, M. G. (2017). A branch and price algorithm to solve the integrated production planning and scheduling in bulk ports. European Journal of Operational Research, 258(3), 926-937. DOI: 10.1016/j.ejor.2016.08.073
73. Shao, X., Li, X., Gao, L., & Zhang, C. (2009). Integration of process planning and scheduling—a modified genetic algorithm-based approach. Computers & Operations Research, 36(6), 2082-2096. DOI: 10.1016/j.cor.2008.07.006

74. Petrović, M., Vuković, N., Mitić, M., & Miljković, Z. (2016). Integration of process planning and scheduling using chaotic particle swarm optimization algorithm. Expert systems with Applications, 64, 569-588. https://doi.org/10.1016/j.eswa.2016.08.019
75. Bertolini, M., Mezzogori, D., & Zammori, F. (2019). Comparison of new metaheuristics, for the solution of an integrated jobs-maintenance scheduling problem. Expert Systems with Applications, 122, 118-136. https://doi.org/10.1016/j.eswa.2018.12.034
76. Liu, X., Ni, Z., & Qiu, X. (2016). Application of ant colony optimization algorithm in integrated process planning and scheduling. The International Journal of Advanced Manufacturing Technology, 84(1-4), 393-404. DOI 10.1007/s00170-015-8145-4
77. Giret, A., Trentesaux, D., & Prabhu, V. (2015). Sustainability in manufacturing operations scheduling: A state of the art review. Journal of Manufacturing Systems, 37, 126-140. https://doi.org/10.1016/j.jmsy.2015.08.002
78. Cheng, C. Y., Lin, S. W., Pourhejazy, P., Ying, K. C., Li, S. F., & Liu, Y. C. (2020). Greedy-Based Non-Dominated Sorting Genetic Algorithm III for Optimizing Single-Machine Scheduling Problem With Interfering Jobs. IEEE Access, 8, 142543-142556. Digital Object Identifier 10.1109/ACCESS.2020.3014134
79. Zuo, Y., Wang, Y., Laili, Y., Liao, T. W., & Tao, F. (2020). An evolutionary algorithm recommendation method with a case study in flow shop scheduling. The International Journal of Advanced Manufacturing Technology, 109(3), 781-796. https://doi.org/10.1007/s00170-020-05471-y
80. Liu, N., Zhang, Y. F., & Lu, W. F. (2020). Energy-efficient integration of process planning and scheduling in discrete parts manufacturing with a heuristic-based two-stage approach. The International Journal of Advanced Manufacturing Technology, 106(5), 2415-2432. https://doi.org/10.1007/s00170-019-04776-x
81. Leng, J., Ruan, G., Jiang, P., Xu, K., Liu, Q., Zhou, X., & Liu, C. (2020). Blockchain-empowered sustainable manufacturing and product lifecycle management in industry 4.0:

A survey. Renewable and Sustainable Energy Reviews, 132, 110112. https://doi.org/10.1016/j.rser.2020.110112

82. Rauch, E., Dallinger, M., Dallasega, P., & Matt, D. T. (2015). Sustainability in manufacturing through distributed manufacturing systems (DMS). Procedia CIRP, 29, 544-549. doi: 10.1016/j.procir.2015.01.069
83. Schinckus, C. (2020). The good, the bad and the ugly: An overview of the sustainability of blockchain technology. Energy Research & Social Science, 69, 101614. https://doi.org/10.1016/j.erss.2020.101614
84. Ko, T., Lee, J., & Ryu, D. (2018). Blockchain technology and manufacturing industry: Real-time transparency and cost savings. Sustainability, 10(11), 4274. https://doi.org/10.1016/j.erss.2020.101614
85. Saberi, S., Kouhizadeh, M., Sarkis, J., & Shen, L. (2019). Blockchain technology and its relationships to sustainable supply chain management. International Journal of Production Research, 57(7), 2117-2135. https://doi.org/10.1080/00207543.2018.1533261.
86. Menezes, G. C., Mateus, G. R., & Ravetti, M. G. (2017). A branch and price algorithm to solve the integrated production planning and scheduling in bulk ports. European Journal of Operational Research, 258(3), 926-937. DOI: 10.1016/j.ejor.2016.08.073.
87. Shao, X., Li, X., Gao, L., & Zhang, C. (2009). Integration of process planning and scheduling—a modified genetic algorithm-based approach. Computers & Operations Research, 36(6), 2082-2096. DOI: 10.1016/j.cor.2008.07.006.
88. Bertolini, M., Mezzogori, D., & Zammori, F. (2019). Comparison of new metaheuristics, for the solution of an integrated jobs-maintenance scheduling problem. Expert Systems with Applications, 122, 118-136. https://doi.org/10.1016/j.eswa.2018.12.034.
89. Giret, A., Trentesaux, D., & Prabhu, V. (2015). Sustainability in manufacturing operations scheduling: A state of the art review. Journal of Manufacturing Systems, 37, 126-140.
90. Cheng, C. Y., Lin, S. W., Pourhejazy, P., Ying, K. C., Li, S. F., & Liu, Y. C. (2020). Greedy-Based Non-Dominated Sorting Genetic Algorithm III for Optimizing Single-Machine Scheduling Problem with Interfering Jobs. IEEE Access, 8, 142543-142556. Digital Object Identifier 10.1109/ACCESS.2020.3014134.
91. Zuo, Y., Wang, Y., Laili, Y., Liao, T. W., & Tao, F. (2020). An evolutionary algorithm recommendation method with a case study in flow shop scheduling. The International Journal of Advanced Manufacturing Technology, 109(3), 781-796. https://doi.org/10.1007/s00170-020-05471-y.

92. Liu, N., Zhang, Y. F., & Lu, W. F. (2020). Energy-efficient integration of process planning and scheduling in discrete parts manufacturing with a heuristic-based two-stage approach. The International Journal of Advanced Manufacturing Technology, 106(5), 2415-2432. https://doi.org/10.1007/s00170-019-04776-x.

93. Srai, J. S., Graham, G., Hennelly, P., Phillips, W., Kapletia, D., & Lorentz, H. (2020). Distributed manufacturing: a new form of localised production?. *International Journal of Operations & Production Management*. https://www.emerald.com/insight/0144-3577.htm

94. Ur-Rahman, N., & Harding, J. A. (2012). Textual data mining for industrial knowledge management and text classification: A business oriented approach. *Expert Systems with Applications*, *39*(5), 4729-4739. https://doi.org/10.1016/j.eswa.2011.09.124

95. Zhang, Jie, Liang Gao, Felix TS Chan, and Peigen Li. (2003) "A holonic architecture of the concurrent integrated process planning system." *Journal of Materials Processing Technology* 139, no. 1-3: 267-272. https://doi.org/10.1016/S0924-0136(03)00233-4

96. Shao, Xinyu, Xinyu Li, Liang Gao, and Chaoyong Zhang. (2009) "Integration of process planning and scheduling—a modified genetic algorithm-based approach." *Computers & Operations Research* 36, no. 6: 2082-2096. https://doi.org/10.1016/j.cor.2008.07.006

97. Choudhary, Alok Kumar, P. I. Oluikpe, Jenny A. Harding, and Patricia M. Carrillo. (2009): "The needs and benefits of Text Mining applications on Post-Project Reviews." *Computers in Industry* 60, no. 9 728-740. https://doi.org/10.1016/j.compind.2009.05.006

98. Mirjalili, S. (2015). Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowledge-based systems*, 89, 228-249. https://doi.org/10.1016/j.knosys.2015.07.006

99. Tang, D., & Dai, M. (2015). Energy-efficient approach to minimizing the energy consumption in an extended job-shop scheduling problem. *Chinese Journal of Mechanical Engineering*, 28(5), 1048-1055. https://doi.org/10.3901/CJME.2015.0617.082

100. Jin, L., Zhang, C., & Fei, X. (2019). Realizing energy savings in integrated process planning and scheduling. *Processes*, 7(3), 120. doi:10.3390/pr7030120

101. C.-Y. Cheng, S.-F. Li, K.-C. Ying, and Y.-H. Liu, (2019)`` Scheduling jobs of two competing agents on a single machine," IEEE Access, vol. 7, pp. 9870298714,. doi 10.1109/ACCESS.2019.2929582

102. K.-C. Ying, S.-W. Lin, and S.-Y. Wan, (2014)``Bi-objective reentrant hybrid flowshop scheduling: An iterated Pareto greedy algorithm," *Int. J. Prod.Res.*, vol. 52, no. 19, pp. 5735_5747. https://doi.org/10.1080/00207543.2014.910627

103. S.-W. Lin and K.-C. Ying, (2015)``A multi-point simulated annealing heuristic for solving multiple objective unrelated parallel machine scheduling problems,"*Int. J. Prod. Res.*, vol. 53, no. 4, pp. 1065_1076. https://doi.org/10.1080/00207543.2014.942011
104. Audet, C., Bigeon, J., Cartier, D., Le Digabel, S., & Salomon, L. (2020). Performance indicators in multiobjective optimization. *European journal of operational research*. https://doi.org/10.1016/j.ejor.2020.11.016.
105. Ishibuchi, H., Imada, R., Setoguchi, Y., & Nojima, Y. (2018). How to specify a reference point in hypervolume calculation for fair performance comparison. *Evolutionary computation*, *26*(3), 411-440. https://doi.org/10.1162/evco_a_00226
106. Haber, R. E., Beruvides, G., Quiza, R., & Hernandez, A. (2017). A simple multi-objective optimization based on the cross-entropy method. *Ieee Access*, *5*, 22272-22281. doi: 10.1109/ACCESS.2017.2764047.
107. Haapala, K. R., Zhao, F., Camelio, J., Sutherland, J. W., Skerlos, S. J., Dornfeld, D. A., ... & Rickli, J. L. (2013). A review of engineering research in sustainable manufacturing. *Journal of manufacturing science and engineering*, *135*(4).
108. Jovane, F., Yoshikawa, H., Alting, L., Boer, C. R., Westkamper, E., Williams, D., ... & Paci, A. M. (2008). The incoming global technological and industrial revolution towards competitive sustainable manufacturing. *CIRP annals*, *57*(2), 641-659.
109. Goldberg, D. E., Korb, B., & Deb, K. (1989). Messy genetic algorithms: Motivation, analysis, and first results. *Complex systems*, *3*(5), 493-530.
110. Srinivas, N., Deb, K., Roy, P. K., & Mehta, S. D. (1995, March). Comparative study of vector evaluated GA and NSGA applied to multiobjective optimization. In *Proceedings of the symposium on genetic algorithms* (pp. 83-90).
111. Özgüven, C., Özbakır, L., & Yavuz, Y. (2010). Mathematical models for job-shop scheduling problems with routing and process plan flexibility. Applied Mathematical Modelling, 34(6), 1539-1548. doi:10.1016/j.apm.2009.09.002
112. Torabi Yeganeh, F., & Zegordi, S. H. (2020). A multi-objective optimization approach to project scheduling with resiliency criteria under uncertain activity duration. Annals of Operations Research, 285(1), 161-196. https://doi.org/10.1007/s10479-019-03375-z
113. Rick, B., Adam, M., Stephen, L., Marty, H., & Brenna, S. (2017). The smart factory; Responsive, adaptive, connected manufacturing.

114. Rauch, E., Dallasega, P., & Matt, D. T. (2016). Sustainable production in emerging markets through Distributed Manufacturing Systems (DMS). Journal of Cleaner Production, 135, 127-138. DOI: 10.1016/j.jclepro.2016.06.106.

115. Varela, L., Araújo, A., Ávila, P., Castro, H., & Putnik, G. (2019). Evaluation of the relation between lean manufacturing, Industry 4.0, and sustainability. Sustainability, 11(5), 1439. https://doi.org/10.3390/su11051439.

116. Fornasiero, R., & Carpanzano, E. (2017). Advances in customer-oriented manufacturing and value chain management. https://doi.org/10.1080/0951192X.2017.1314016

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

118. Silvestri, L., Forcina, A., Introna, V., Santolamazza, A., & Cesarotti, V. (2020). Maintenance transformation through Industry 4.0 technologies: a systematic literature review. Computers in Industry, 123, 103335. https://doi.org/10.1016/j.compind.2020.103335

119. Abeyratne, S. A., & Monfared, R. P. (2016). Blockchain ready manufacturing supply chain using distributed ledger. International Journal of Research in Engineering and Technology, 5(9), 1-10. https://doi.org/10.15623/ijret.2016.0509001

120. Giret, A., Trentesaux, D., & Prabhu, V. (2015). Sustainability in manufacturing operations scheduling: A state of the art review. Journal of Manufacturing Systems, 37, 126-140.

121. Manupati, V. K., Deo, S., Cheikhrouhou, N., & Tiwari, M. K. (2012). Optimal process plan selection in networked based manufacturing using game-theoretic approach. International Journal of Production Research, 50(18), 5239-5258. https://doi.org/10.1080 /00207543.2012.682181.

122. Reddy, M. S., Ratnam, C. H., Agrawal, R., Varela, M. L. R., Sharma, I., & Manupati, V. K. (2017). Investigation of reconfiguration effect on makespan with social network method for flexible job shop scheduling problem. Computers & Industrial Engineering, 110, 231-241. https://doi.org/10.1016/j.cie.2017.06.014.

123. Afshin Mansouri, S., & Aktas, E. (2016). Minimizing energy consumption and makespan in a two-machine flowshop scheduling problem. Journal of the Operational Research Society, 67(11), 1382-1394. https://doi.org/10.1057/jors.2016.4.

124. Yan, J., Li, L., Zhao, F., Zhang, F., & Zhao, Q. (2016). A multi-level optimization approach for energy-efficient flexible flow shop scheduling. Journal of Cleaner Production, 137, 1543-1552. https://doi.org/10.1016/j.jclepro.2016.06.161.

125. Tavakkoli-Moghaddam, R., Ranjbar-Bourani, M., Amin, G. R., & Siadat, A. (2012). A cell formation problem considering machine utilization and alternative process routes by scatter search. Journal of Intelligent Manufacturing, 23(4), 1127-1139. DOI 10.1007/s10845-010-0395-2.

126. Seada, H., & Deb, K. (2014). U-NSGA-III: A unified evolutionary algorithm for single, multiple, and many-objective optimization. COIN report, 2014022.