Computers & Industrial Engineering 192 (2024) 110240

Contents lists available at ScienceDirect

Computers & Industrial Engineering

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

Meta-heuristic algorithms for integrating manufacturing and supply chain functions

Onur Canpolat [a,1,*], Halil Ibrahim Demir [b,2], Caner Erden [c,d,3]

[a] Department of Software Engineering, Faculty of Engineering, Firat University, Elazig, Türkiye
[b] Department of Industrial Engineering, Faculty of Engineering, Sakarya University, Sakarya, Türkiye
[c] Department of Computer Engineering, Faculty of Technology, Sakarya University of Applied Science, Sakarya, Türkiye
[d] AI Research and Application Center, Sakarya University of Applied Sciences, Sakarya, Türkiye

ARTICLE INFO

Keywords:
Integrating manufacturing and supply chain functions
Process planning
Scheduling
Due date assignment
Delivery
Meta-Heuristic algorithms

ABSTRACT

This study proposes an integrated framework that incorporates essential manufacturing and supply chain functions. These functions encompass process planning, scheduling, due-date assignment, and delivery optimization. The objective of this integrated approach is to achieve multiple benefits, including balanced workload distribution, enhanced company performance, generation of more realistic planning schedules, and ultimately, the achievement of shorter due dates. As a result, the overall efficiency of operations is substantially improved, with approximately a 50 % increase over isolated function management. Additionally, the isolated integration of the delivery function within systems comprising three integrated functions has been found to improve efficiency by 18%. The study employs various heuristic techniques, including genetic algorithms, simulated annealing, random search, hybrid search, and evolutionary strategy, to assess the optimal solution method and rules for these functions. The Taguchi technique is employed to ascertain the optimal values for critical parameters, such as population size, mutation rate, crossover points, and random search rate. Among the solution methods investigated, genetic algorithms consistently yielded superior results Additionally, the weighted slack rule consistently exhibited notable effectiveness compared to other due-date assignment rules. Similarly, the savings algorithm outperformed other delivery optimization rules. However, it is important to note that among the scheduling rules evaluated, none has emerged as dominant.

1. Introduction

A thriving manufacturing ecosystem relies on the seamless integration of critical functions, each of which is pivotal for achieving operational excellence. In manufacturing systems, where effectiveness and optimization are paramount, the convergence of process planning, scheduling, due date assignment (DDA), and delivery functions has emerged as a critical driver for enhancing operational efficiency and increasing customer satisfaction. Although each function has individual significance, its combined impact within a unified system is substantially greater. Process planning, one of these essential functions, involves systematically assessing, determining, and organizing the processes and materials used throughout the manufacturing of products. It encompasses tasks such as defining operational routes and selecting appropriate machines. However, it is crucial to note that process planning comes with associated costs, and planners must carefully consider these expenses. Implementing manual and independent process planning on the shop floor (SF) can lead to chaos, reduce performance, and increase dissatisfaction. The only tangible benefits of manual process planning reside in its modest cost implications and flexibility, which allow for expeditious and facile system modifications (Scallan, 2003).

Scheduling is critical function in the manufacturing industry that directly impacts various operational aspects (Zhao et al., 2021). Optimizing the scheduling process has numerous benefits, including increased efficiency, heightened product quality, and meeting customer demands. The primary objective of production scheduling is to

* Corresponding author.
E-mail addresses: onurcanpolat@firat.edu.tr (O. Canpolat), hidemir@sakarya.edu.tr (H. Ibrahim Demir), cerden@subu.edu.tr (C. Erden).
[1] https://orcid.org/0000-0003-4790-1685.
[2] https://orcid.org/0000-0003-1949-9676.
[3] https://orcid.org/0000-0002-9955-2256.

https://doi.org/10.1016/j.cie.2024.110240

Available online 22 May 2024

maximize resource allocation and job sequencing, even when faced with varying requirements and challenges across different industries. Scheduling is not only a cost-saving function but also contributes to revenue generation by reducing expenses, minimizing delays, and stabilizing machine workloads (Chen & Hall, 2008).

DDA is of pivotal importance in manufacturing. Prioritizing customer satisfaction is paramount when assigning due dates to individual tasks. Setting overly distant due dates can harm a business's reputation and lead to customer loss. This situation can also result in due-dates and earliness penalties, with early task completion leading to unnecessary stockpiling and holding costs. Conversely, unrealistic due-dates can result in increased tardiness costs, damaged reputation due to unfulfilled promises, and additional expenses, such as compensatory price reductions offered to dissatisfied customers. The Just-in-Time (JIT) principle aims to complete work precisely on the due date (Gordon et al., 2002). Timely delivery enhances customer satisfaction and bolsters a company's reputation.

The literature reveals a variety of sub-integrations of four key functions in production and logistics. Notably, the integration of process planning and scheduling (IPPS) has been rigorously investigated by Phanden et al., (2011 and further discussed by Li and Gao (2020). Similarly, SWDDA has been explored by Vinod and Sridharan (2011) and by Koulamas and Kyparisis (2022) or while SWDWA has been addressed by Janiak et al. (2015) and by Wang and Li (2019). IPODS problems have been studied by Fu et al. (2017), PTSP problems were examined by Lacomme et al. (2016) and Karaoğlan and Kesen (2017). Furthermore, the integration of process planning, scheduling, and due-date assignment (IPPSDDA) was analyzed by Demir, Kökçam, and Erden (2023) and (2021). Dynamic versions of IPPSDDA were introduced by Demir and Erden (2020) and further elaborated by Erden et al. (2023), which integrated three functions. According to our comprehensive review of the existing literature, this study represents the first instance of integrating all four functions, marking a significant advancement in the field.

When reviewing all sub-integrations, it becomes apparent that each emphasizes the necessity and value of integration. Notably, the integrated functions of process planning, scheduling, and delivery are independently classified as NP-hard problems, as detailed in the literature (Demir, Kökçam, & Erden, 2024; Liu, Li, Gao, & Fan, 2022; Tarhini, Danach, & Harfouche, 2022). Consequently, the complexity of all integrated problems escalates, with the integration of the four functions, as considered in this study, representing the most complex scenario. The integration of these functions marks a pioneering research area and holds the potential to deliver the highest global return. This comprehensive integration encapsulates all the benefits and challenges previously identified in individual sub-integrations, and even more.

In the literature concerning scheduling with due-date assignment (SWDDA) and scheduling with due-window assignment (SWDWA), due dates are defined as the expected or planned completion times for production. These due dates can also indicate the specific times when products are loaded onto vehicles or when goods are prepared for delivery. In this study, the due date is the promised delivery date for the products to the customer. Moreover, in numerous vehicle routing problems (VRP), integrated production and outbound distribution scheduling problems (IPODS), and production and transportation scheduling problems (PTSP), delivery due dates or due windows are not optimized as part of the problem formulation; instead, these problems are typically solved using pre-determined delivery due dates or due windows. The SWDDA problem, explored extensively by scholars such as (Gordon et al. (2002), Keskinocak & Tayur (2004)), and Vinod and Sridharan (2011), and further discussed in Demir, Kökçam, & Erden (2024), is presented in its most general form. This configuration assigns a specific due date to tasks, defining the target completion time. Conversely, the SWDWA problem, as delineated in studies by Janiak et al. (2015) and again in Demir et al. (2024), involves assigning a due window rather than a precise due date. This variant provides a time frame within which the task should be completed, offering a more flexible approach to scheduling. This distinction highlights the evolution of scheduling methodologies to accommodate different operational flexibilities and constraints, thereby addressing a broader spectrum of industrial and logistical challenges.

In order to delve into the specific benefits and challenges of this study, a systematic analysis of the current sub-integrations is crucial. IPPS has shown multiple advantages, such as increased profitability and enhanced product quality. Furthermore, the efficient utilization of resources leads to improved delivery times. Moreover, the seamless implementation of alternative production plans in the SF enables quick responses to unforeseen situations, thus highlighting the essential role of IPPS in optimizing the coordination between product design and practical manufacturing processes (Liu et al., 2022; Phanden et al., 2011; Yu et al., 2018).

The complexity of IPPS is greatly intensified due to the NP-hard classification of the job shop scheduling problem. As integration levels increase, the complexity of manufacturing flexibility also grows, making it harder to address these challenges. When the complexity of problems escalates, precise solutions often become unachievable within restricted time limits. Therefore, the literature mostly focuses on the success of metaheuristic methods in dealing with these challenging issues (Liu et al., 2022). Without IPPS, the achievement of a functional Computer-Integrated Manufacturing System (CIMS) that integrates diverse manufacturing phases into a unified system is impeded. Furthermore, IPPS is poised to significantly enhance flexibility, adaptability, agility, and global optimization within distributed and collaborative manufacturing environments. This underscores the urgent need for continued research into IPPS, as Li and Gao (2020) posited in their work. Such endeavors are critical for advancing the field and optimizing manufacturing processes to meet contemporary demands.

In this research, the delivery function represents the fourth and critical component integrated into the problem, potentially offering the most substantial contribution to the overall system performance. While single-item and single-customer deliveries may not necessitate complex routing strategies, more general delivery scenarios often evolve into a type of VRP. In the context of SWDDA and SWDWA, delivery is considered at the point when products are ready to be dispatched. However, in integrated production and outbound distribution scheduling problems (IPODS) or production and transportation scheduling problems (PTSP), delivery may be handled by a third-party logistics (3PL) company through outsourcing. Typically, delivery entails distributing products directly to the customers' locations, which inherently involves solving a VRP subproblem.

The principal target is to handle assignments on due dates and scheduling concurrently. Nonetheless, two methodologies have been posited in academic literature due to the complexity of integrated solutions. These methodologies are outlined as follows: 1) Addressing both functions concurrently by merging their objectives. 2) Recognizing the challenge of addressing both functions simultaneously, one objective function is addressed while integrating the other as a constraint in the problem. Due date assignment significantly influences performance, as performance is contingent upon both scheduling and due date assignment. While tardiness traditionally incurs penalties in performance evaluations, the advent of Just-in-Time (JIT) philosophy has led to penalties for both earliness and tardiness simultaneously. The importance of due date assignment amplifies as production transitions from make-to-stock (MTS) to make-to-order (MTO) types (Keskinocak & Tayur, 2004).

The primary focus of this study is to incorporate the quadruple function, a domain that has not been explored previously. The delivery function is introduced into the IPPSDDA problem as the fourth function for the first time, resulting in the attainment of more comprehensive and successful solutions. This integration encompasses all the challenges and benefits delineated thus far. IPPSDDA, the core manufacturing function, is a highly effective approach for organizations seeking to maximize customer satisfaction compared to the traditional independent approach. Despite the extensive academic research concerning integration, these functions commonly operate sequentially and independently. Integration reduces costs and shortens delivery times, increasing customer satisfaction when effectively coordinated. An integrated manufacturing system and scheduling decisions are derived from process-planning outputs, as shown in Fig. 1. This integration fosters a balanced workload, enhances operational efficiency, enables realistic plans, shortens timelines, and enhances adaptability to change.

In the manufacturing and logistics landscape, integrating operational functions has emerged as a key strategy to enhance efficiency and customer satisfaction. Traditional methodologies often treat process planning, scheduling, due-date assignment, and delivery in isolation. However, this approach fails to capture the interdependencies between these critical functions. Our study introduces an innovative framework that integrates these four functions within a manufacturing system to provide a comprehensive solution leveraging their interconnectedness. This paper explores the challenges involved in such integration but aims to address them by adopting an integrated approach for more effective management. The study utilizes a range of advanced metaheuristic methodologies, including genetic algorithms (GA), simulated annealing (SA), and other hybrid search techniques, to navigate the increased complexity and provide viable solutions within practical timeframes.

The novelty of this research lies in its attempt to merge these functions and prioritize customer-centric outcomes, ensuring that delivery dates and production schedules are optimized to enhance customer satisfaction and operational performance. This paper details the development of a sophisticated mathematical model that supports this integration, presenting a significant advancement in the field of Computer-Integrated Manufacturing Systems (CIMS). Through this integrated framework, the study seeks to deliver a significant contribution to the literature and practice, demonstrating the potential for increased global performance and customer-oriented manufacturing strategies.

This study establishes an integrative framework between the delivery function and process planning, scheduling, and due-date assignment. Contrary to the SWDDA sub-integration in the literature, the due date here does not indicate when the production is completed but when the goods will be delivered to the customer at the door. The intention is to facilitate the integration of production and supply chain processes, consequently enhancing efficiency and productivity. This study aims to demonstrate the viability of integration, compare the advantages offered by integrated solutions in contrast to sequential and independent approaches, and evaluate the impact of partial integration by explicitly focusing on the delivery function.

Acknowledging that customers who engage in order placement exhibit heterogeneity in their importance to a company is essential. Customers who consistently order products or place substantial order volumes are typically regarded as more valuable than others. One of the important aspects of the present study is to analyze the repercussions of assessing customers based on their distinctive levels of importance rather than uniformly treating all customers on global performance. Specifically, this study aims to investigate the influence of integrating customer importance considerations into DDA, scheduling, and order delivery processes. Notably, this study diverges from previous research by including penalties for earliness, tardiness, and due date-related based on customer-specific importance weights, thereby advancing the existing understanding of this issue. The main contributions and novelty of this paper can be summarized as follows:

- The integration of the delivery function and delivery date with process planning and scheduling functions represents a novel aspect of this study.
- Unlike traditional approaches that assign due dates for scheduling, this study assigns doorstep delivery dates to customers.
- Production planning and delivery operations deviate from the conventional practice of adhering to predetermined external delivery dates found in existing literature. Instead, delivery dates are internally optimized within the problem framework.
- The delivery dates, scheduling, and actual delivery operations are assigned based on customer weights, reflecting their relative importance. Priority is given to important customers to enhance overall weighted performance.
- Process planning, scheduling, and vehicle routing represent inherently combinatorial and NP-hard problems. Additionally, IPPS, SWDDA, IPODS, and PTSP problems exhibit increased complexity, with the IPPSDDA problem presenting an even greater level of intricacy. Furthermore, the IPPSDDAD problem under investigation within this study emerges as the most complex challenge.
- Even relatively less intricate problems and sub-integrations, such as scheduling, vehicle routing, IPPS, SWDDA, IPODS, PTSP, and IPPSDDA, often necessitate the utilization of metaheuristic methodologies for resolution. Accordingly, this study utilizes standard metaheuristics to tackle the most intricate problem, IPPSDDAD. Additionally, a complex mathematical model has been devised and presented to address this intricate problem in parallel with this research.

2. Related works

Tan and Khoshnevis (2000) conducted an empirical investigation of an integrated process planning and scheduling (IPPS) framework, highlighting the advantages of integration and delving into the various techniques employed by researchers in this domain. Sobeyko and Mönch (2017) implemented IPPS in a large-scale flexible job-shop production environment with different product trees and routes. They considered the weighted total tardiness as a performance measure and used mixed-integer programming to solve the problem. Petrović et al. (2016) assessed a new heuristic algorithm, antlion optimization, for IPPS and demonstrated its feasibility. Márquez and Ribeiro (2022) prepared a review paper on the problems in SF and flow manufacturing environments between 2000 and 2022. Yang et al. (2022) considered an IPPS to minimize the total completion time for single-machine parallel batching with different job families. To our knowledge, most IPPS studies in the literature presume that customers have equal significance levels. This study emphasizes that each customer may have different degrees of importance.

The emergence of new operations management concepts, such as JIT and supply chain management, has increased interest in SWDDA. Zhao and Tang (2014) considered a single-machine scheduling problem in which the processing time of a job depends on its position in the queue.

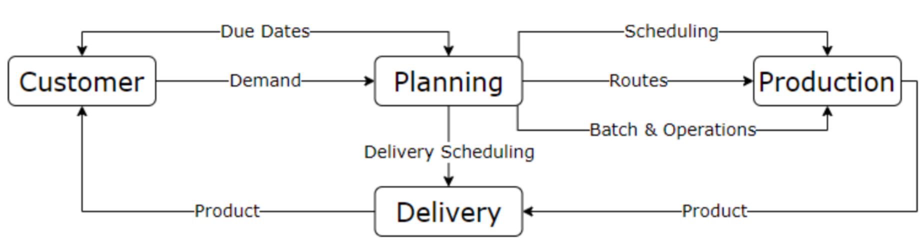

Fig. 1. Integrated manufacturing system.

They have an objective function that includes the cost of changing the due date, the cost of discarded jobs, and the sum of jobs that finish early. Three different DDA rules are used. The customer determines a reasonable and acceptable due date, and it is assumed that no penalty cost is incurred unless the due date is set above this limit. Pan et al. (2023) investigated the single-machine SWDDA problem with earliness, tardiness, and due-date costs. Arik (2023) studied a SWDDA in which jobs are given different earliness and tardiness weights. Qian and Han (2022) studied a single-machine scheduling problem with delivery time and deteriorating jobs under three forms of due dates. Yin et al. (2021) studied a SWDDA in which an agent represented each function. Xu et al. (2021) studied a multitask SWDDA in a batch manufacturing environment in which each job is processed individually. Mor et al. (2021) developed a set of universal heuristic algorithms to solve single-machine scheduling problems with resource-dependent processing times.

In some studies where scheduling and delivery are integrated, products are delivered by the manufacturer and shipped to the customer's doorstep (Garcia & Lozano, 2004; Tonizza Pereira & Seido Nagano, 2022), In some studies, delivery optimization was performed until the products are loaded onto the vehicle and the vehicle's path was not examined (Li et al., 2017). Some studies use third-party logistics (3PL) companies for delivery planning (Han et al., 2019). Berghman et al. (2023) reviewed studies on integrated production and distribution scheduling (IPDS). Chen and Li (2020) studied an IPDS problem in which each product is processed on specialized machines. It is possible to obtain various products in order using three heuristic algorithms. Huang et al. (2023) studied the IPDS problem in a batch manufacturing environment with multiturn vehicles and due-window assignments. Solina & Mirabelli (Solina & Mirabelli, 2021) studied the IPDS problem in a natural food business, considering the perishability factor. Tarhan and Oğuz (2021) studied an IPDS problem that aimed to maximize net revenue using a local search scheme.

Moons et al. (2017) provided a comprehensive assessment of the existing literature on IPDS, highlighting that prior research has predominantly focused on relatively simplistic scenarios, wherein each order comprises a single process. Many studies have primarily

Table 1
Categorical review of studies in the literature.

Problem	Reference	MM	GA	ACO	SA	TS	MIP	PSO	DA	Other
IPPS	Lee and Kim (2001)		✓							
	Morad and Zalzala (1999)		✓							
	Zhang and Wong (2015)		✓							
	Shen and Yao (2015)	✓								
	Leung et al. (2010)			✓						
	Guo et al. (2009)							✓		
	Özgüven et al. (2010)	✓								
	Moon et al. (2008)	✓					✓			
	Zhang and Wong (2018)			✓						
	Ba et al. (2018)							✓		
	Luo et al. (2017)		✓							
	Petrović et al. (2016)							✓		
	Jin et al. (2016)									✓
	Yu et al. (2015)		✓					✓		
	Mohapatra et al. (2015)		✓							
	Wang et al. (2014)			✓						
	Chu et al. (2015)						✓			
SWDDA	Chen et al. (2007)	✓					✓			
	Zhang and Wu (2012)		✓							
	Yue and Zhou (2021)									✓
	Chen et al. (2023)								✓	✓
	Zhang et al. (2022)								✓	
	Atsmony and Mosheiov (2024)				✓					
	Shabtay et al. (2022)	✓								
	Arik et al. (2022)								✓	
IPDS	Grigoreva (2020)									✓
	Shahin Moghadam et al. (2014)			✓	✓					
	Hou et al. (2022)									✓
	Liu et al. (2021)	✓								
	Long et al. (2022)							✓		
	Luo et al. (2023)									✓
	Ullrich (2013)		✓							
	Zhan and Wan (2018)					✓				
	Wang et al. (2020)	✓					✓			
	Ghannadpour and Zarrabi (2018)		✓							
	Yağmur and Kesen (2021)						✓			
	Chen et al. (2009)	✓								
	Bo et al. (2021)								✓	
	Reiter et al. (2011)						✓			
IPPSDDA	Erden et al. (2019)		✓		✓	✓				
	Demir et al. (2015)		✓							
	Demir and Phanden (2019)		✓		✓					
	Demir et al. (2021)				✓					
	Demir and Erden (2020)			✓						
	This Study (IPPSDDAD)		✓		✓					

investigated single or parallel machine environments, often employing the same parallel machines. However, considering the widespread adoption of production environments characterized by multiple production levels, such as job shops or flow shops for mass production, there is a need to integrate these environments with vehicle routing, presenting a promising avenue for future research. Yağmur and Kesen (2023) analyzed the IPDS problem within a job shop environment, allowing multiple customer visits. Table 1 presents an overview of pertinent studies addressing integration and outlines the methods used in these studies such as a mathematical model (MM), Ant Colony Optimization (ACO), Tabu Search (TS), Mixed Integer Programming (MIP), Particle Swarm Optimization (PSO), and the author's proprietary algorithm, referred to as DA.

Although IPPSDDA has the potential to generate highly efficient outcomes, it has yet to receive significant attention in the academic literature due to its inherent challenges and complexities. Initially, Demir and Taskin (2005) investigated this topic within the scope of their doctoral thesis. Although they penalized earliness and tardiness, this study extends this approach by penalizing due-dates and earliness. Furthermore, this thesis determined due dates without considering customer weights. In contrast, this study assigns critical customers closer due dates and schedules important ones at earlier time slots.

Erden (2019) dynamized IPPSDDA with stochastic and dynamic job arrivals in PhD thesis. In that thesis, jobs can arrive at the SF at any time, according to an exponential distribution. Demir and Erden (2020) solved the dynamic IPPSDDA problem using the ACO algorithm. In this study, the IPPSDDA problem in the literature is extended with a delivery function, and earliness, tardiness, and due-date costs are included in the objective function with customer weights. Two different routes are used in process planning: four rules are used in the DDA, ten are used in scheduling, and nine are used in delivery.

In reviewing the literature, notable research has been conducted on integrated process planning, scheduling, due-date assignment, and delivery within manufacturing systems, but a notable research gap persists. Although previous studies have addressed individual components of this integrated system, such as process planning or scheduling, few have holistically tackled the integration of all these functions simultaneously. Moreover, existing research predominantly focuses on simplistic scenarios or specific production environments, leaving a significant void in addressing the complexities of real-world manufacturing systems characterized by diverse production levels and intricate logistics. Therefore, the research presented in this paper fills this critical gap by proposing an integrated framework that comprehensively addresses process planning, scheduling, due-date assignment, and delivery within a manufacturing system, thereby offering a novel approach to optimize production operations and meet customer demands effectively.

Extending the framework of IPPSDDA by integrating the delivery function, this study explores an aspect of integration that, to the best of our knowledge, has not been studied before. Additionally, by incorporating due date costs–elements that have found minimum representation in prior studies- into the performance function and aspects of earliness and tardiness, this study aims to devise solutions in line with the JIT principle.

3. Materials and methods

3.1. Problem definition

This study is about integrated process planning, scheduling, delivery date assignment, and delivery (IPPSDDAD), a new topic that has never been studied in the literature. Process planning, scheduling, vehicle routing issues, and combinatorial and NP-Hard problems have been widely studied. Starting from the last decades of the previous century, these functions have been integrated dually and triple. IPPS, SWDDA, IPODS, and PTSP problems have been commonly studied in the literature. Recently, the problem of Process planning, scheduling, and Due-date assignment integration, namely IPPSDDA, has started to be studied, but the (IPPSDDAD) issue has not been addressed yet, and this study is the first study to address this problem and solve problems on this subject.

In this study, J jobs will be divided into P lots, and each batch will have a maximum of b jobs. Each batch will be processed and scheduled on the floor according to its priorities. M machines on the shopfloor process these jobs. Each job j has total R alternative routes, and each route has O operations. Each job is denoted by j while m signifies the machine. In this context, p denotes the party of the job among P parties, j represents the job among J jobs, r signifies the selected route of the job j among R alternative routes, o represents the current operation of job j according to route r among O operations, and m denotes the machine that operates operation o among M machines. In this study, in the mathematical model, the vehicle is assumed to be ready when necessary, and in the simulation, a single vehicle is assumed to make multiple trips. It is accepted that J jobs are waiting when the simulation started, and a static shop floor simulation is performed.

The following assumptions are accepted in this study.

1. At the outset, the study presupposes the presence of J jobs comprising multiple operations.
2. Delivery dates are determined concurrently with the problem as integrated into the problem, and considering these dates, the jobs are produced and delivered to the customers, and the total performance measure is calculated.
3. When multiple jobs wait for the same machine, the job selection adheres to dispatching rules.
4. Completed operations are distributed in batches according to selected delivery rules.
5. The duration of each operation conforms to a normal distribution and is characterized by non-deterministic behavior.
6. Each job follows a unique process route.
7. Predecessor operations must be completed before subsequent operations can commence.
8. Jobs may necessitate operations to be executed on diverse machines.
9. Both machines and vehicles will not encounter breakdowns.
10. Each order belongs to a different customer.
11. Each customer has a different importance and, consequently, a weight.
12. Jobs not completed within a working day are postponed to the next day.
13. Machines are different, and each machine can perform certain tasks.
14. The arrival time of each job on the shop floor is predetermined (deterministic).
15. There is a single vehicle capable of making multiple tours.
16. Routes for each job contain an equal number of operations.
17. There are J independent jobs and M machines.
18. The processing time of an operation on alternative machines is predefined.
19. Operations cannot be interrupted while being executed.
20. Each operation can only be performed on one machine; similarly, a machine can only handle one operation at a time.
21. In batch delivery, it is assumed that the vehicle sets out with a specific number of orders, neither below nor above.
22. Loading products at the shop floor during delivery and unloading them from the vehicle upon reaching the customer occurs within the delivery time.
23. If the last batch of jobs does not fill the vehicle's capacity, the remaining portion from the next group is transferred.
24. The vehicle does not leave the shop floor until its capacity is filled.
25. Each route starts and ends at the shop floor.

3.2. Mixed-integer linear programming model

This study explores a job-shop-style manufacturing environment, wherein jobs are presumed to be known and available at the onset. The selection of process plans, optimal batching, job scheduling, assignment of delivery dates, and determination of vehicle routing are all concurrently established through the mathematical model. Upon completion of a production batch, it is loaded onto a vehicle and dispatched to the customers. Vehicles are presumed to be available as required.

The mathematical model employed in our study is structured upon the notations delineated in Table 2, with the performance function and constraints elucidated in Table 3. This approach draws upon the foundational study of (Özgüven et al., 2010), who developed comprehensive mathematical models for flexible job-shop scheduling problems (FJSPs), incorporating both routing and sequencing sub-problems, as well as process plan flexibility (FJSP-PPFs). Below model includes process plan selection, optimal batching, machine load balancing, minimizing total machining time, optimizing delivery date of every job, job shop scheduling and vehicle routing of each batch. Objective function minimizes makespan of all job, makespan of every batch and completion time of every job. Total transportation cost and total weighted earliness, tardiness and delivery date penalties are minimized.

D_{pj} (in this case due dates are given to the model as an outside parameters)

Variables
$Y_{pjrop'j'r'o'm}, X_{pjrom}, V_{jkp}, Z_{pjr} = 0 \text{ or } 1$

$P_{pjrom}, S_{pjrom}, C_{pjrom}, L_m, L_{max}, C_{pj}, C_p^{max}, C_{max}, e_{pj}, l_{pj}, a_{pk}, E_{pj}, T_{pj}, D_{pj} \geq 0$

D_{pj} (in this case due dates are optimized with the problem)

Parameters
$t_{oj}, t_{jk}, b, S_j, C^t_{0j}, C^t_{jk}, tc, w_j, D_{pj}$

Table 2
Summary of notations used in the mathematical model.

Notation	Description	Notation	Description
X_{pjrom}	*pjro*(batch, job, route, operation) 1 if processed on machine *m*, 0 otherwise	J	Jobs ($j \in J$) in every batch $n = J+1$ depot + customers in a batch
P_{pjrom}	Processing time of *pjro* operation on machine *m*	P	Parties ($p \in P$)
S_{pjrom}	Start time of *pjro* operation on machine *m*	K	Customers ($k \in K$)
C_{pjrom}	End time of *pjro* operation on machine *m*	M^c	Machines ($m \in M^c$)
Z_{pjr}	1 if job *j* is selected to batch *p* and uses route *r*, 0 otherwise	$\mathbb{M}$	A big number
V_{0kp}	1 if customer *k* belonging to batch *p* was the first to arrive from the shop floor, 0 otherwise (0 represents shop floor)	b	Batch size
V_{jkp}	In batch *p*, 1 if we go from customer *j* to customer *k*, 0 otherwise	tc	Transport cost per km
C_{pj}	Completion time of job *j* of batch *p*	S_j	service time at customer *j*
C_p^{max}, C_{max}	Makespan of batch *p*, makespan of all jobs	t_{0k}, t_{jk}	Arrival time from shop floor (0) to customer *k*/ from customer *j* to *k*
D_{pj}	Delivery-date of job *j* in party *p*	C^t_{0j}, C^t_{jk}	Distance from shop floor (0) to customer *j*/from customer *j* to *k*
E_{pj}, T_{pj}	Earliness and Tardiness times regarding the delivery-date	L_m, L_{max}	The total load of machine *m* and load of the machine with maximum load
L_e, L_T	Earliness and tardiness penalties per unit time	a_{pk}	Arrival time at customer *k* belonging to party *p*
$Y_{pjrop'j'r'o'm}$	1 if operation P_{pjrom} precedes operation $P_{p'j'r'o',m}$; otherwise, 0.	L	Delivery-date penalty per unit time
w_j	Weight or importance of customer *j*		

3.3. Problem structure and specifications

The investigated problem encompasses four distinct SFs, specifically SF1, SF2, SF3, and SF4, each characterized by differing job quantities of 25, 50, 75, and 100, respectively. Notably, each job undergoes processing via two alternative production routes, encompassing three discrete operations. Another salient aspect of this study pertains to the varying levels of customer importance, stratified into four distinct categories: especially important, important, moderately important, and less important, corresponding to the relative importance scores of 2.5, 1, 0.5, and 0.33, respectively. Therefore, it is imperative to prioritize customers of higher importance at each stage of the production process. Customers are equally distributed to each category.

Process plans should be the primary focus of SF integration. The process plan defines the tasks that must be accomplished in the SF, specifying the execution methods, tools required, time frame for completion, and other relevant details. The sample processing plan is presented in Table 4.

The jobs are grouped into batches, with each job assigned to only one batch and each batch containing a maximum of five jobs (customers). The batches are created by ordering the jobs in SF based on their batch value (BV), which is determined by the processing time for each job in SF (p_j), importance of the customer (w_j), and the customer's distance from SF (d_{0j}), all in descending order of priority. The BV value is calculated using Eq. (22) below.

$$BV = \left(\sum p_j + d_{0j}\right) * \frac{1}{w_j} \tag{22}$$

Once the BV values of the jobs have been established, they are arranged in ascending order. This sorting process is based on the rationale that a lower BV value corresponds to closer proximity to the SF, association with a more significant customer, or increased importance due to a shorter production time. The allocation of important jobs to priority batches substantially contributes to overall performance, as it guarantees that more important customers are served with heightened priority.

To ensure successful delivery, it is crucial to have accurate knowledge of each customer's location. This study defines customer locations as random points on a coordinate system, with SF being the system's center (0,0). The distances between each customer and the SFs are calculated according to the rectilinear distance between them by using Eq. (23).

$$d_{ij} = \left|(y_j - y_i)\right| + \left|(x_j - x_i)\right| \tag{23}$$

In this equation, d_{ij} represents the distance between two points and x and y represent the coordinates of those points. Table 5 shows the distance matrix.

Table 3
The constraints of the integrated problem and their explanations.

No	Equation		Explanation
(1)	$MinZ = \left(M^{c}L_{max} + \sum_{m=1}^{M} L_m\right) + \left(C_{max} + \sum_{p=1}^{P} C_p^{max} + \sum_{p=1}^{P}\sum_{j=1}^{J} C_{pj}\right) + \left(\sum_{p=1}^{P}\sum_{j=0,j\neq k}^{J}\sum_{k=0}^{J} tcC_{jk}^{t}V_{jkp}\right) + \left(\sum_{j=1}^{J} w_j E_{pj} L_e + \sum_{j=1}^{J} w_j T_{pj} L_T + \sum_{j=1}^{J} w_j D_{pj} L\right)$		In the performance function, in the first parenthesis (the load of the maximum loaded machine + the total load on the machines is tried to be minimized. In the second parenthesis (makespan of all job, makespan of every batch and completion time of every job are tried to be minimized). In the third bracket (transportation costs are minimized). In the fourth bracket (Weighted Earliness + Tardiness + Length of delivery date are penalized)
(2)	$\sum_p \sum_r Z_{pjr} = 1$	$\forall j$	If $Z_{pjr} = 1$ job j is in batch p and uses route r. Job j can belong to only one party and only one of the routes can be selected.
(3)	$\sum_j \sum_r Z_{pjr} \leq b$	$\forall p$	There can be a maximum of b (batch size) jobs in a batch.
(4)	$X_{pjrom} \leq P_{pjrom}$	$\forall p, \forall j, \forall r, \forall o, \forall m$	If P_{pjrom} is zero then X_{pjrom} should be zero.
(5)	$X_{pjrom} = Z_{pjr}$	$\forall p, \forall j, \forall r, \forall o,$ m for $\forall o$	If $Z_{pjr} = 1$ then operations o of job j, which is in batch p and uses route r, should be processed on machines m.
(6)	$\sum_p \sum_j \sum_r \sum_o X_{pjrom} P_{pjrom} = L_m$	$\forall m$	L_m returns the total workload on machine m.
(7)	$L_m \leq L_{max}$	$\forall m$	L_{max} the load of the maximum loaded machine which is the highest machine load among the loads of all the machines.
(8)	$\sum_r Z_{pjr} \leq \sum_{k=0} V_{jkp}(8)$ $\sum_r Z_{pkr} \leq \sum_{j=0} V_{jkp}(8')$	$\forall j, \forall p(8)$ $\forall k, \forall p(8')$	These two constraints are coordination constraints between party assignment, job shop scheduling and vehicle routing. Suppose any location from customer j in party p is visited in the vehicle routing section. In that case, then customer j belongs to party p in the party assignment, scheduling and vehicle routing section. In the scheduling section of the indices, jobs start from $j = 1$ to the number J jobs. In vehicle routing j starts from 0 where 0 represents the shop floor and the number of customers ends at J. $8'$) If customer k in party p is reached from anywhere in the vehicle routing section, then customer k belongs to party p in the party assignment, scheduling and vehicle routing section.
(9)	$S_{pjrom} + C_{pjrom} \leq X_{pjrom}\mathbb{M}$	$\forall p, \forall j, \forall r, \forall \mathrm{o}, \forall \mathrm{m}$	If a job j belongs to batch p, uses route r, is processed by operation o and is processed on machine m, then that operation of that job can have a start and end time on that machine. Shortly if X_{pjrom} is 1 then S_{pjrom}, C_{pjrom} start time and completion time can have value. These constraints are written only when P_{pjrom} values are positive. For zero values no such constraints are required to write.
(10)	$C_{pjrom} \geq S_{pjrom} + P_{pjrom} - \left(1 - X_{pjrom}\right)\mathbb{M}$	$\forall p, \forall j, \forall r, \forall \mathrm{o}, \forall \mathrm{m}$	If an operation is processed on a machine, the end time of that job on that machine must be greater than the start time + the processing time of that operation. These constraints are written only when P_{pjrom} values are positive. For zero values no such constraints are required to write.
(11)	$S_{pjrom} \geq C_{p'j'r'o'm} - \left(Y_{pjrop'j'r'o'm}\right)\mathbb{M}(11)$ $S_{p'j'r'o',m} \geq C_{pjrom} - \left(1 - Y_{pjrop'j'r'o',m}\right)\mathbb{M}(11')$	$\forall p, \forall j, \forall r, \forall \mathrm{o}, \forall p', \forall j', \forall r', \forall \mathrm{o}', \forall \mathrm{m}$	These constraints are either, or constraints. The P_{pjrom} operation is before or after the $p'j'r'o'm$ operation. Either the start of the operation $pjrom$ must be before the operation $p'j'r'o'm$ (jobs processed on the same machine m, then the binary variable $\left(Y_{pjrop'j'r'o',m}\right) = 1$), or the beginning of the $p'j'r'o'm$ operation should be before the $pjrom$ operation, then the binary variable $\left(Y_{pjrop'j'r'o',m}\right) = 0$). These constraints are written only when P_{pjrom} or $P_{p'j'r'o',m}$ values are positive. For zero values no such constraints are required to write.
(12)	$S_{pjrom} \geq C_{pjr,\mathrm{o}-1,\mathrm{m}}$	$\forall p, \forall j, \forall r, \forall \mathrm{o} - \{o_{(f)}\}$	The start of the $pjro$ operation must be after the end of the $pjr(o-1)$ operation. Because $(o-1)$ is the operation before the o operation in the r^{th} route. These constraints are written starting from the second operation (Except $o_{(f)}$ first operation)
(13)	$C_{pj} = \sum_r \sum_{o=o_{(l)}} C_{p,j,r,o_{(l)},m}$	$\forall p, \forall j$	C_{pj} that is, the completion time of job j selected for batch p, must be equal to the last processing time last operations, $o_{(l)}$ of all routes of this job. Among all last operations' $o_{(l)}$ completion times on all machines $C_{p,j,r,o_{(l)},m}$ only one route, one last operation and one machine take value, the others are zero.
(14)	$C_p^{max} \geq C_{pj}$	$\forall p, \forall j$	The makespan of batch p is greater than the completion time of all jobs in that batch. The makespan of all jobs is greater than the makespan of all parties.
(15)	$C_{max} \geq C_p^{max}$	$\forall p$	
(16)	$a_{pk} = C_p^{max} + t_{ok}$ if $V_{0kp} = 1$	$\forall k, \forall p$	Arrival at customer k is the makespan of the lot this customer is in the delivery time from the shop floor to this customer (if this customer is the first delivery in that lot)
(17)	$a_{pk} = a_{pj} + S_j + t_{jk}$ if $V_{jkp} = 1$	$\forall k, \forall p, \forall j, j \neq k$	If customer k is not the first delivery, then if customer j is visited before this customer, then arrival time at this customer is equal to the arrival time at customer j + service time at customer j + transportation time between j and k.
(18)	$E_{pj} = max\{D_{pj} - a_{pj}, 0\}$	$\forall j, \forall p$	Early arrival according to the delivery date is found by subtracting the arrival time to customer j from the delivery date. The delay is found by subtracting the delivery date from the arrival time to customer j. Here, the delivery date of job j (D_{pj}) can be variable and optimized with the mathematical model or can be given form outside as parameters to the model and model is solved according to the given delivery dates. The model works for both cases.
(19)	$T_{pj} = max\{a_{pj} - D_{pj}, 0\}$	$\forall j, \forall p$	
(20)	$\sum_{k=0,k\neq j} \sum_{p=1} V_{jkp} = 1$ $\sum_{j=0,j\neq k} \sum_p V_{jkp} = 1$	$\forall j, j \neq 0\ \forall k, k \neq 0$	These are vehicle routing constraints. The first constraint tells us that in each batch (i.e., each vehicle), we need to go from each customer j to another customer k or the shop floor 0. The second constraint tells us that in each batch (that is, in each vehicle), we need to come from another customer, j or 0, from the shop floor to a k customer.
(21)	$u_j - u_k + n \times V_{jkp} \leq n - 1$ $2 \leq u_j \leq n$	$\forall j \neq 0, \forall k \neq 0, j \neq k \forall \mathrm{j}, \mathrm{j} \neq 0$	These two SEC constraints are sub-tour elimination constraints. We start from the shop floor and end on the shop floor. If we do not count start and end, every customer in that batch should be visited once in one unique sequence between 1(shop floor) and $n + 1$(shop floor). that means the sequence of customer j must be unique and should be in between $2 \leq u_j \leq n$.

The chromosome comprises three essential genes: due-date assignment, scheduling, and delivery. The first gene had four rules, whereas the others contained ten and nine rules. Fig. 2 illustrates this problem in its simplest terms.

Regardless of the rules selected for each function (DDA, scheduling, or delivery) in the chromosome, the problem is resolved by integrating these rules. Each rule had a corresponding value for the relevant gene. Each chromosome is formed by combining the functional and route genes. The chromosomal structure is shown in Fig. 3. The number of jobs in a shop influences the length of the chromosomes.

Table 6 shows a part of the sample chromosomes of a population. The length of the chromosomes varies according to the number of jobs on the shop floor. Since there are two routes for each job, the route genes have a value of 0 or 1.

For each rule, a corresponding value is assigned to the pertinent gene. The rules based on these functions are listed in Table 7, where d_i

Table 4
Sample process plan.

Jobs	Routes	Operation 1		Operation 2		Operation 3		CustomerImportance
		Process Time	Machine	Process Time	Machine	Process Time	Machine	
J1	R0	6	1	5	2	9	2	2.50
	R1	8	2	6	2	5	1	
J2	R0	9	2	8	1	3	1	0.50
	R1	8	1	4	1	3	2	
J3	R0	3	1	3	1	9	2	2.50
	R1	4	1	7	2	6	1	
J4	R0	9	2	5	2	7	1	1.00
	R1	6	2	7	1	5	1	

Table 5
Distance matrix.

Distance	SF	J1	J2	J3	J4	J5	J6	J7	J8	J9	J10	J11	J12	J13
SF	−1	44	77	80	69	5	22	44	35	11	22	25	32	49
J1	44	−1	121	46	63	45	66	18	43	37	60	61	72	51
J2	77	121	−1	93	80	76	55	121	112	88	61	60	49	126
J3	80	46	93	−1	35	75	90	64	89	83	58	105	112	97
J4	69	63	80	35	−1	64	79	63	78	72	47	94	101	86
J5	5	45	76	75	64	−1	21	45	36	12	17	30	37	50
J6	22	66	55	90	79	21	−1	66	57	33	32	15	22	71
J7	44	18	121	64	63	45	66	−1	25	33	60	61	72	33
J8	35	43	112	89	78	36	57	25	−1	24	51	52	63	14
J9	11	37	88	83	72	12	33	33	24	−1	27	28	39	38
J10	22	60	61	58	47	17	32	60	51	27	−1	47	54	65
J11	25	61	60	105	94	30	15	61	52	28	47	−1	11	66
J12	32	72	49	112	101	37	22	72	63	39	54	11	−1	77
J13	49	51	126	97	86	50	71	33	14	38	65	66	77	−1

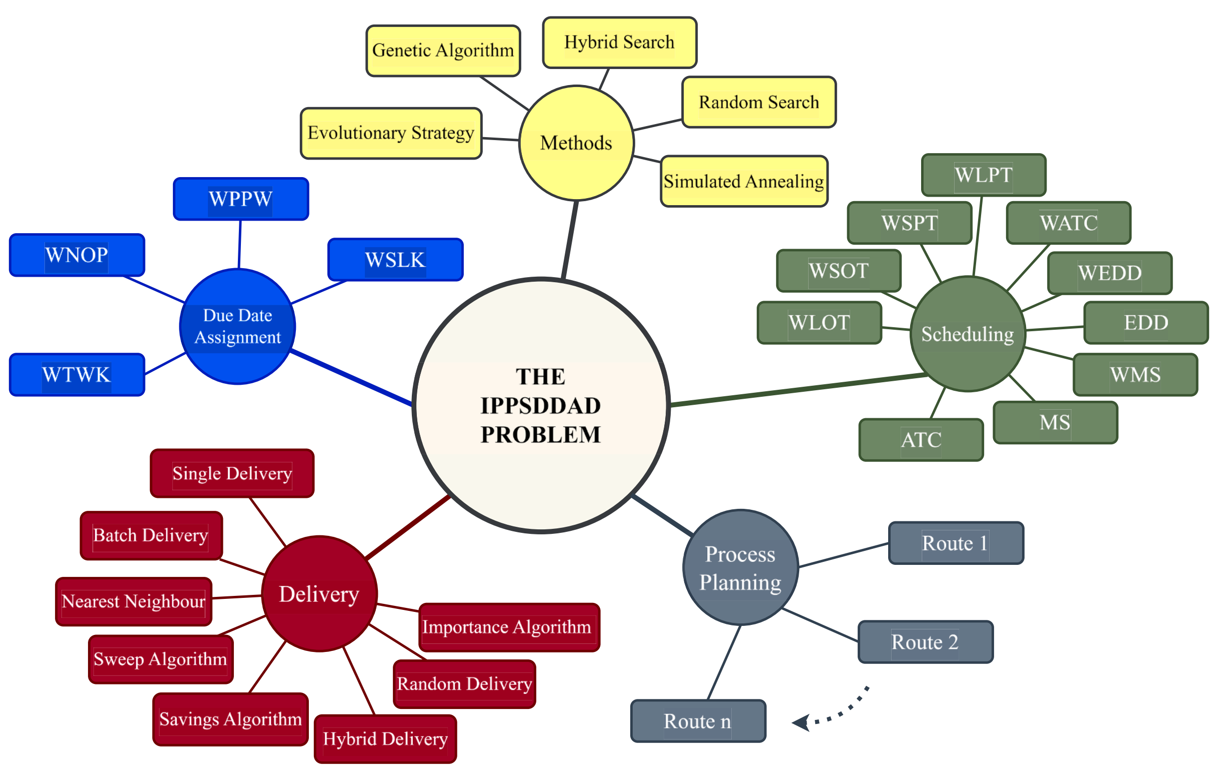

Fig. 2. The IPPSDDAD problem.

represents the delivery date, w_i denotes the customer weight, N_i signifies the number of operations for job i, q_i denotes the slack value, p_i represents the total process time for job i, k denotes the predetermined constant value. Z_1 and Z_2 are determined inversely proportional to customer weights.

According to the approach outlined before, rather than handling the production and the delivery functions independently, they will be structured to work together and serve a unified purpose, as shown in Fig. 4.

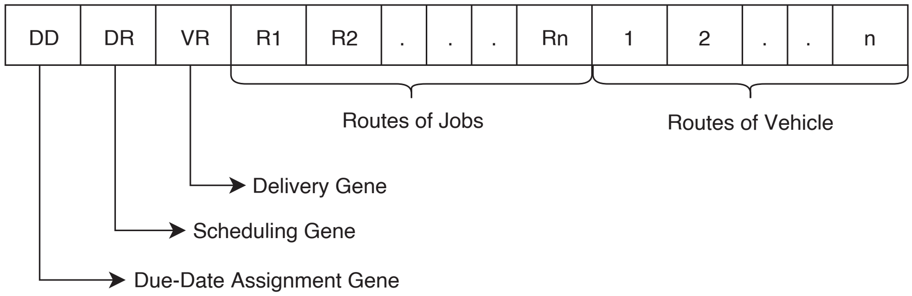

Fig. 3. Chromosome structure.

Table 6
A sample chromosome.

Chromosomes		DD	DR	VR	J1	J2	J3	J4	J5	J6	J7	J8	J9	J10	J11	J12	J13	J14	J15	J16	
Chromosome 1	[	0	5	3	0	1	1	1	1	0	0	0	0	1	1	1	0	1	0	0	]
Chromosome 2	[	1	1	5	1	0	1	1	1	1	0	0	0	0	1	0	0	1	1	0	]
Chromosome 3	[	0	8	7	1	0	1	0	1	1	1	0	0	1	1	1	0	1	0	0	]
Chromosome 4	[	3	3	1	0	0	0	1	1	1	0	1	1	1	1	0	1	0	1	1	]
Chromosome 5	[	0	9	1	0	0	1	1	1	1	1	1	0	1	0	1	0	1	0	1	]
Chromosome 6	[	2	5	4	1	1	1	1	0	1	1	1	1	0	0	1	0	1	1	0	]
Chromosome 7	[	1	2	3	1	0	0	1	0	0	0	0	1	0	0	0	1	0	1	1	]
Chromosome 8	[	2	1	2	0	0	0	0	0	0	1	0	1	0	0	1	0	0	1	1	]
Chromosome 9	[	1	7	1	0	1	0	0	0	1	1	0	1	1	1	0	0	0	0	1	]
Chromosome 10	[	1	0	0	1	1	0	0	1	1	1	1	0	1	0	0	0	1	1	1	]

Table 7
Gene values of the rules.

No	DDA rules		Dispatching rules		Delivery type
0	WSLK	$d_i = p_i + q_i{}^*Z_i$	WSPT	$I_i = \frac{w_i}{p_i}$	Single delivery
1	WPPW	$d_i = p_i{}^*k^*Z_1 + q_i{}^*Z_2$	WSOT	$I_i = \frac{w_i}{p_{ij}}$	Batch delivery
2	WNOP	$d_i = N_i{}^*k^*Z_i$	WLOT	$I_i = \frac{p_{ij}}{w_i}$	Nearest neighbor
3	WTWK	$d_i = p_i{}^*k^*Z_i$	WLPT	$I_i = \frac{p_i}{w_i}$	Savings algorithm
4			WATC	$I_i = \frac{w_i}{p_i}e^{\left(\frac{max(slack,0)}{K\overline{p}}\right)}$	Sweep algorithm
5			ATC	$I_i = \frac{1}{p_i}e^{\left(\frac{max(slack,0)}{K\overline{p}}\right)}$	Random delivery
6			MS	$I_i = -(slack)$	Hybrid delivery
7			WMS	$I_i = -(slack)w_i$	Priority Algorithm 1
8			EDD	$I_i = \frac{1}{d_i}$	Priority Algorithm 2
9			WEDD	$I_i = \frac{w_i}{d_i}$	

3.4. Performance criterion

The objective is a function based on the delivery time. This function penalizes the promised due date, tardiness, and earliness, if any. Tardiness (T_{pj}) is calculated using Eq. (24), and earliness (E_{pj}) is calculated using Eq. (25).

$$T_{pj} = max(a_{pj} - D_{pj}, 0) \tag{24}$$

$$E_{pj} = \max(D_{pj} - a_{pj}, 0) \tag{25}$$

The penalties are determined using Eq. (26) and (27), where w_j is the customer's weight, a_{pj} is the completion time, and D_{pj} is the delivery date.

$$P_{Ej} = w_j{}^*(5 + 4^*(\frac{E_{pj}}{480})) \tag{26}$$

$$P_{Tj} = w_j{}^*(10 + 8^*(\frac{T_{pj}}{480})) \tag{27}$$

The performance function incorporates a promised due date, which aligns with the emphasis of the JIT production philosophy on timely job completion (Gordon et al., 2002). The philosophy prioritizes meeting due dates precisely and considers earliness or tardiness unfavorable. Because the problem prefers lower performance function values, assigning jobs closer to their order time minimizes the penalties calculated using Eq. (28) for the given due-date length. The total penalty (P_j) for a job (denoted by tp) is the sum of the P_{Dj}, P_{Ej}, P_{Tj} as formulated in Eq. (29). The performance criterion (PC) in this study minimizes the sum of the penalties calculated for all jobs, as shown in Eq. (30).

$$P_{Dj} = w_j{}^*(8^*(\frac{D_{pj}}{480})) \tag{28}$$

$$P_j = P_{Dj} + P_{Ej} + P_{Tj} \tag{29}$$

$$PC = \sum\nolimits_{j=1}^{n} P_j \tag{30}$$

3.5. Meta-heuristic algorithms

Meta-heuristic algorithms and their applications can be classified into various domains based on the problem they aim to solve and the specific characteristics of the algorithm. It spans various domains, from optimization and searches to machine learning, data mining, decision-making, and control systems, offering versatile solutions to complex problems in various fields. For a second classification approach, they can be classified based on the underlying strategy they employ to explore and exploit the solution space. One common classification scheme categorizes *meta*-heuristics into single solution-based and population-based algorithms. Single solution-based *meta*-heuristics operate with a single candidate solution at a time. These algorithms iteratively explore the solution space by making incremental modifications to the current

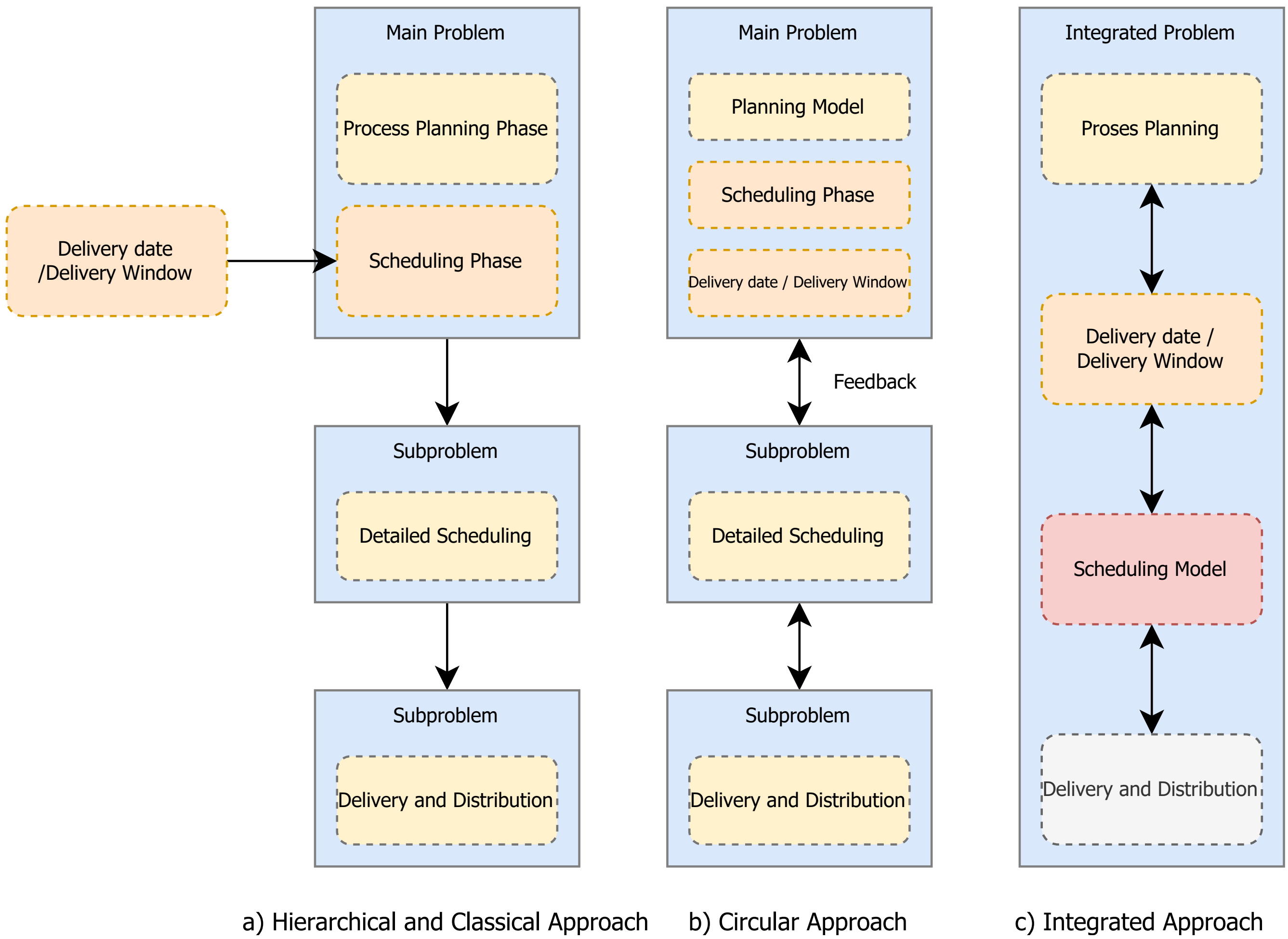

Fig. 4. Solution strategies of production functions.

solution, aiming to improve its quality. Several single solution-based *meta*-heuristic algorithms include hill climbing, simulated annealing, and tabu search. Single solution-based algorithms are simpler and require fewer computational resources, while population-based algorithms offer better exploration capabilities and can handle complex solution spaces more effectively. However, both types of *meta*-heuristics offer advantages depending on the problem characteristics.

Another characteristic of *meta*-heuristic algorithms is whether they are directed or undirected to search for better solutions. Directed search algorithms such as GA, ES, and SA use the best solutions found so far to find better solutions. Undirected search algorithms such as RS do not use the best solutions found, and, in every iteration, they try brand new solutions in the solution space. RS algorithm scans the solution space extremely fast at the beginning, but the chance of finding better solutions decreases as iterations go on. On the other hand, directed searches use the best solutions to find better solutions, which is the power of directed search algorithms. A hybrid search is applied to combine the power of the undirected RS algorithm and the power of directed GA. The first five iterations are applied using the RS algorithm at the beginning, and the remaining iterations are applied using GA.

3.6. Solution representation and decoding

To address this problem, a range of optimization algorithms is employed, including SA, GA, ES, HS, and RS. SA focuses on optimizing an individual chromosome throughout each iteration, as opposed to operating with populations, which is a practice observed in certain alternative algorithms. The SA algorithm begins by establishing the temperature and last temperature and the cooling coefficient. Subsequently, a chromosome is selected as the initial solution, and its performance is evaluated and recorded as both the current and optimal solutions. This chromosome is then subjected to mutation to generate a neighboring solution, which is then stored as the new solution. The performance of the new solution is assessed and compared with that of the current solution.

If an enhancement in the solution is observed, the new solution is deemed as the current solution (Cura, 2008). However, if no improvement is evident, the acceptance of the new solution relies on the outcome of the acceptance probability (p_{accept}). Once the p_{accept} is computed, a random number is generated and compared with p_{accept}. If the p_{accept} surpasses the random number, the new solution is accepted. Otherwise, the current solution is retained, the temperature is lowered, and a new iteration is initiated. This process persists until the temperature value drops below the final temperature preset. Fig. 5 shows the SA algorithm.

Rather than exploring the entire solution space, the GA expedites the process by testing a subset of existing solutions, yielding suitable solutions in a reduced timeframe (Ismail et al., 2022). The GA selects ideal chromosomes from an ample space by employing solution values known as fitness functions (Lin & Jou, 2000). The search for improved solutions centers on the best solutions identified thus far within each iteration of the GA. During each iteration, a specific number of chromosomes is chosen from the parent population, and a cross-population is subsequently generated via the crossover operator. Subsequently, the mutation operator selects a designated number of chromosomes for mutation, creating a mutation population. The crossover and mutation rates in the

Algorithm 1 Simulated Annealing

Begin
Define temperature (T), last temperature (LT) and cooling coefficient (C)
Generate a chromosome (current)
Evaluate the performance of the chromosome $(p_{current})$
While $(T \geq LT)$
    Mutate the current chromosome (new)
    Evaluate the mutation chromosome (p_{new})
     Set due dates by the DDA rule
     Schedule by scheduling rule
     Route by vehicle routing rule
    If $(p_{new} < p_{current})$
      Current chromosome = New chromosome
    Else
      Generate a random number (RN)
      $p_{accept} = exp\left(\frac{p_{current} - p_{new}}{T}\right)$
      If $(p_{accept} > RN)$
        Current chromosome = New chromosome
      Else
      $T = T * C$
End

Fig. 5. Steps of the SA.

GA are 60 % (6 out of 10 chromosomes) and 40 % (4 out of 10 chromosomes), respectively. After these processes, chromosomes exhibiting superior performance are more likely to be selected (Göçken et al., 2018). Fig. 6 illustrates the steps involved in the GA.

In each iteration, 6 out of 10 chromosomes belonging to the population are randomly selected for crossover so that they are not repeated. The multi-point crossover performed in the study is shown in Fig. 7 and Fig. 8.

In each iteration, a subset of chromosomes, specifically four out of the total ten, are selected for mutation. During the mutation process, a particular number of genes, optimized through Taguchi methods, are modified within each chromosome. Notably, the mutation is executed using two distinct approaches. This differentiation arose from the fact that genes other than the first three within the chromosome exhibit binary characteristics, taking either a value of 1 or 0. Conversely, the first three genes within the chromosome encompass various numerical values.

For instance, the second gene is associated with scheduling and offers a choice from ten different scheduling methods. Should this gene be selected for mutation, a random selection is made from the permissible values (excluding the current value). In contrast, the mutation operator functions differently for other genes, employing a mechanism where the value of the gene transitions to 1 if it is previously 0 or 0 if it is formerly 1. An illustration of the mutation process is presented in Fig. 9.

ES is an approach that aims to explore and identify improved solutions within the neighbors of reasonable solutions, drawing inspiration from evolutionary theory. In contrast to GA, which employs crossover and mutation operators, ES relies exclusively on the mutation operator. The iterative process commences by concatenating the chromosomes from the current population (initial population in the first iteration) into an array. Subsequently, the fitness function values denoting penalties are computed for each chromosome, arranged in ascending order, and used to derive the selection probability for each chromosome based on this order.

Next, a chromosome is randomly chosen for mutation using the selection probabilities and subjected to the mutation operation. The mutation process is executed on a total of 10 chromosomes. Notably, the same chromosome may be re-selected for mutation based on its selection probabilities, ensuring that superior-quality chromosomes are granted multiple opportunities for mutation. The resulting mutated chromosomes are collected and stored in the mutation array.

The fitness function values for the chromosomes within both the original and mutation population arrays are computed. These chromosomes are subsequently ranked in ascending order based on their fitness function values. Up to the extent of the population size, the top-ranking chromosomes are then transferred to the subsequent iteration as the best chromosomes. This iterative process is repeated for the designated number of iterations.

The RS algorithm randomly generates a population comprising ten chromosomes. Subsequently, the performance of the generated chromosomes is assessed. The top ten chromosomes exhibiting the most favorable performance from the iterations are selected for subsequent

Algorithm 2 Genetic Algorithm

Begin
Define variables and parameters.
Set initial values of variables
Generate the initial population (10 chromosomes)
Evaluate and rank the population
Set due dates by the DDA rule
Schedule by scheduling rule
Route by vehicle routing rule
Evaluate the population
Rank the population
While *(number of iterations* $\leq$ *100) or (time* $\leq$ *2000)*
Perform crossover and mutation
Evaluate the crossed population
Set due dates by the DDA rule
Schedule by scheduling rule
Route by vehicle routing rule
Evaluate the population
Rank the population
Evaluate the mutation population
Set due dates by the DDA rule
Schedule by scheduling rule
Route by vehicle routing rule
Evaluate the population
Rank the population
Update main population
number of iterations = number of iterations + 1
End

Fig. 6. Steps of the GA.

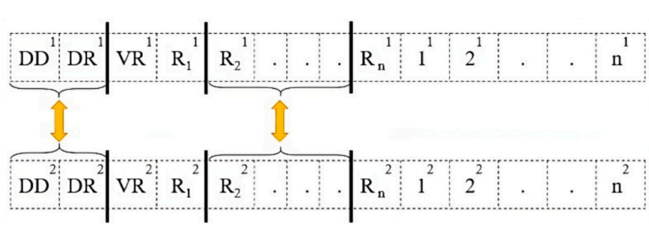

Fig. 7. Chromosomes before crossover.

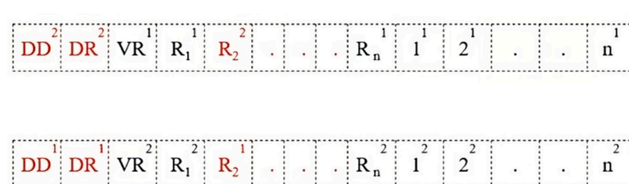

Fig. 8. Chromosomes after crossover.

iterations. Notably, the RS algorithm operates for the same number of iterations as GA.

The HS algorithm is obtained by integrating the GA and RS algorithms. 5 out of 100 iterations employed the RS algorithm, whereas the remaining 95 are conducted using the GA. Ten chromosomes are generated in the initial five iterations, and their performance is evaluated. At the sixth iteration, the top ten chromosomes from the RS algorithm become the starting population for the GA. The GA lasted for 95 iterations, commencing with the updated population. Through this

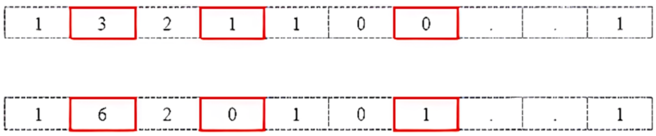

Fig. 9. Mutation process representation.

process, the best-performing chromosome is determined. To ensure fairness and comparability, identical initial populations are employed for ES and GA. Moreover, an equivalent number of iterations is implemented for all methods. Additionally, the population size is consistently maintained throughout all population-based techniques.

3.7. Taguchi design of experiments

The design of experiments (DOE) aims to obtain the optimum output with a minimum number of experiments without requiring the testing of all parameters and levels of these parameters (Taguchi, 1986). When confronted with situations in which exhaustive exploration of the entire solution pool is challenging, a precious approach involves conducting a limited number of experiments using orthogonal arrays. This method allows for examining parameter effects on the solution without compromising the underlying structure and output of the problem. Table 8 presents the relevant parameters utilized in the problem and the corresponding orthogonal arrays required by the number of levels for each parameter.

This study examined the effects of four different parameters consisting of three levels. The Minitab Statistical Software is used for the DOE. The results are analyzed based on the signal-to-noise ratio. The parameters and levels used in DOE are listed in Table 9. The DOE methodology suggests the application of L9 orthogonal arrays, where a 4-parameter experimental design encompasses three levels. Significantly, while conducting 81 experiments would be necessary to obtain a comprehensive understanding of the impact of the parameters on the problem, only nine experiments are conducted using DOE. Table 10 provides complete information about the experiments and their corresponding results.

The parameter values that provide the signal-to-noise ratio and the minimum penalty values in the 95 % confidence interval are taken as the basic values for all shop floors in the study. A graph of the signal-to-noise ratio is shown in Fig. 10.

Based on the findings presented in Fig. 10, it can be observed that among the four parameters analyzed about the problem, the “population size” of 10 demonstrates the most optimal performance. Similarly, for the parameter “number of crossover points”, a value of 1 yields the most effective outcome. Furthermore, the “RS rate” of 0.1 and a parameter value of 13 for the “number of genes to mutate” are found to achieve the most efficient performance. Consequently, employing a population size of 10 chromosomes, implementing a single-point crossover approach, maintaining an RS ratio of 0.1, and mutating 13 genes across all SFs is recommended.

4. Experimental results and discussions

To solve the problem, a computer code is developed in Python using

Table 8
Taguchi orthogonal arrays.

Taguchi orthogonal array		Number of parameters						
		2	3	4	5	6	7	8
Number of Levels	2	L4	L4	L8	L8	L8	L8	L12
	3	L9	L9	L9	L18	L18	L18	L18
	4	L16	L16	L16	L16	L32	L32	L32
	5	L25	L25	L25	L25	L25	L50	L50

Table 9
Parameters and levels for DOE.

Population size (PopSize)	Number of crossover points (No. of CP)	Random search(RS) Rate	Mutation gene count(No. of Mut.)
6	1	5 %	7
10	2	10 %	10
16	3	15 %	13

PyCharm IDE on a computer with an Intel(R) Core(TM) i7-4700HQ processor with 2.40 GHz and 16 GB RAM. NumPy, random and math libraries are used. An experimental analysis assessed the impact of integrating the four distinct functions, commencing with a non-integrated SF (SIRO-RDM). Subsequently, various levels of integration are established and compared with the four integrated SFs. Solution methods are employed to obtain the SF results. The iteration numbers are standardized to facilitate precise evaluation and comparison, and all random numbers used in the software are held constant. Detailed information on the level of integration is presented in Table 11.

The integration levels contain 50 jobs, and their performance without the solution method (raw) and their solution with GA are examined. Table 12 presents the comparative results.

The GA presents a highly productive solution across all levels of integration. At the integration level of SIRO-RDM (Level 1), wherein each function operates independently without explicit rules, the GA remarkably demonstrates a 45 % improvement. Moreover, the GA has significantly improved at the Level4 integration level, representing complete integration. However, it is important to note that this improvement ratio is expected to increase with the number of iterations. As the integration level advances, the problem exhibits the capacity to obtain superior solutions. Fig. 11 illustrates the performance of different integration levels.

Empirical observations demonstrate that, as the level of integration increases, it leads to superior outcomes compared to irregular and random configurations because each function’s degree of customer importance becomes more pronounced and structured. Furthermore, the empirical evidence suggests that the highest integration level, Level4, outperforms the other levels. Table 13 presents the recovery rates across the various integration levels. Additionally, incorporating rules into a problem that contains customer importance enables a marked enhancement in the global solution.

Table 13 clearly indicates a 35 % improvement at the Level2 integration level compared with the SIRO-RDM (Level1) integration level. This signifies the significant impact of integrating even just delivery in a rule-based manner on the problem’s solution. Furthermore, at the Level3 integration level, characterized by random delivery, rule-based due-date assignment, and scheduling, a 41 % enhancement is observed for Level1. Notably, this represents an 9 % improvement compared with the Level2 integration level.

If one bifurcates the integration levels into two distinct groups, Level1 and Level2, and Level3 and Level4, it will facilitate a more comprehensive understanding of the implications of integrating the delivery function. Specifically, when all functions are devoid of rules (Level1), the mere introduction of the delivery function augmented with rules (Level2) enhances the performance by 35 %. Similarly, the level at which all functions are integrated (Level4) outperforms its counterpart by 18 % when delivery is conducted randomly (Level3). The findings demonstrate that incorporating the delivery function into the overarching system consistently yields superior and more efficient global solutions. Table 14 presents an overview of the advantages linked to the integration of delivery at the Level1, with the exclusion of delivery rules [0], [1], and [5] due to their anticipated unfavorable outcomes resulting from their inherent characteristics.

It is observed that integrating delivery rules into a randomized system (Level1) improves performance by at least 30 percent and at most

Table 10
Results of Taguchi DOE.

PopSize	No. of CP	RS rate	No. of Mut.	Result1 (tp)	Result2	Result 3	Result 4	Result 5
6	1	0.05	7	461,246	478,445	480,336	461,198	463,879
6	2	0.10	10	462,006	462,010	463,236	468,583	472,510
6	3	0.15	13	462,017	466,127	482,565	462,317	461,274
10	1	0.10	13	484,428	487,287	493,000	479,137	473,283
10	2	0.15	7	458,143	462,542	482,598	463,207	464,843
10	3	0.05	10	464,062	485,606	461,948	463,934	466,986
16	1	0.15	10	470,752	479,047	464,493	463,239	469,345
16	2	0.05	13	472,148	468,667	463,263	495,874	486,440
16	3	0.10	7	481,104	465,255	456,803	462,042	480,364

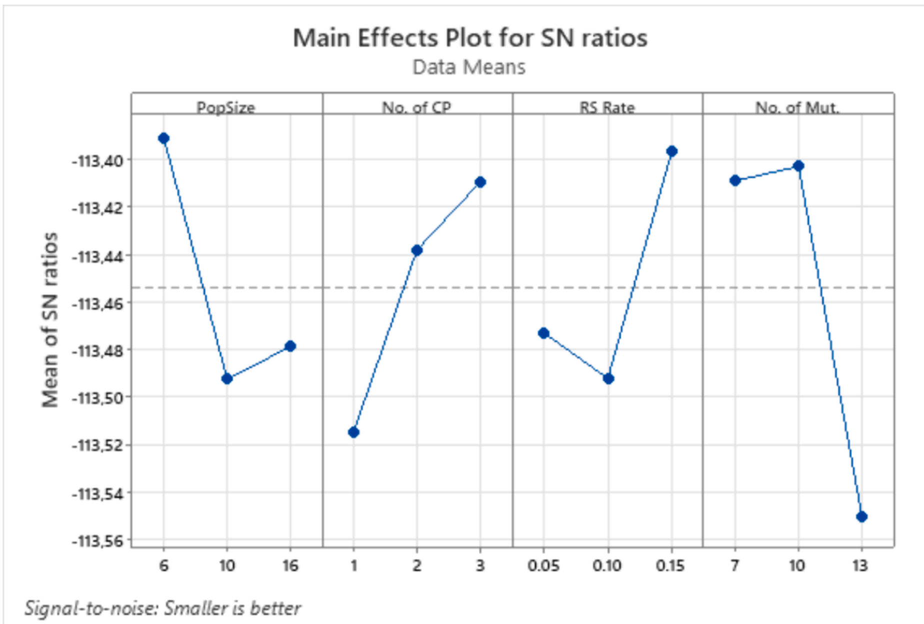

Fig. 10. Taguchi signal-to-noise ratio results.

Table 11
Integration levels.

Integration Level	Due Date Assignment	Scheduling	Delivery
Level1	RDM	SIRO	DIRO
Level2	RDM	SIRO	Rule-based
Level3	Rule-based	Rule-based	DIRO
Level4	Rule-based	Rule-based	Rule-based

Table 12
Performance of integration levels.

Integration Level	Raw Performance	GA Performance	Improvement Rate
Level1	1011714,0 tp	555319,0 tp	45 %
Level2	655708,0 tp	463779,0 tp	29 %
Level3	595048,2 tp	503735,0 tp	15 %
Level4	489319,0 tp	473841,2 tp	3 %

37 percent. Among the delivery rules, the savings algorithm yielded the best results. The substantial improvement distinctly illustrates the impact of integrating delivery into the problem. Specifically, concerning the Level1 level where none of the functions are integrated, performing delivery exclusively within a specific rule demonstrates a minimum overall performance improvement of 30 %. Table 15 presents the results for the four SFs in the problem.

According to the conducted analyses, the GA demonstrated optimal performance in two of the four simulated SFs (SF2 and SF3). ES is the most effective algorithm for SF1, whereas RS yielded the best global solution for SF4. It should be noted that GA began to achieve superior results with a growing number of jobs. For instance, despite trailing behind the ES in SF1, which comprised 25 jobs, it improved by 0.4 % in SF2, housing 50 jobs. Moreover, when the job count reached 75, GA widened the performance gap with the nearest algorithm to 2.3 %. These data suggest that GA's performance improves proportionately with increased job numbers.

However, an anomaly is observed in SF4, which contained 100 jobs. The primary attribute of this irregularity is the large number of iterations required. By solving a considerable problem encompassing 100 jobs, it is inferred that exploring the expansive solution space in only 100 iterations diminished the efficiency of the results. To investigate this further, SF4 underwent 4000 iterations utilizing identical data. Table 16 presents the results of the extended analysis.

As evidenced by the analysis, augmenting the number of iterations leads to a marked enhancement in the GA performance relative to the other algorithms. Given that more jobs engender a more expansive solution space, further exploration favors a global solution. Moreover, alterations are implemented in the chromosomes responsible for rendering the optimal results obtained throughout the study, followed by examining the consequent effect on the results. This process allows for assessing the influence of rules on global performance.

Table 17 presents an analysis of the performance of the DDA rules for each of the four SFs, explicitly focusing on the chromosomes that delivered the best performance with the GA.

An apparent prevalence of the WSLK rule is evident across all four

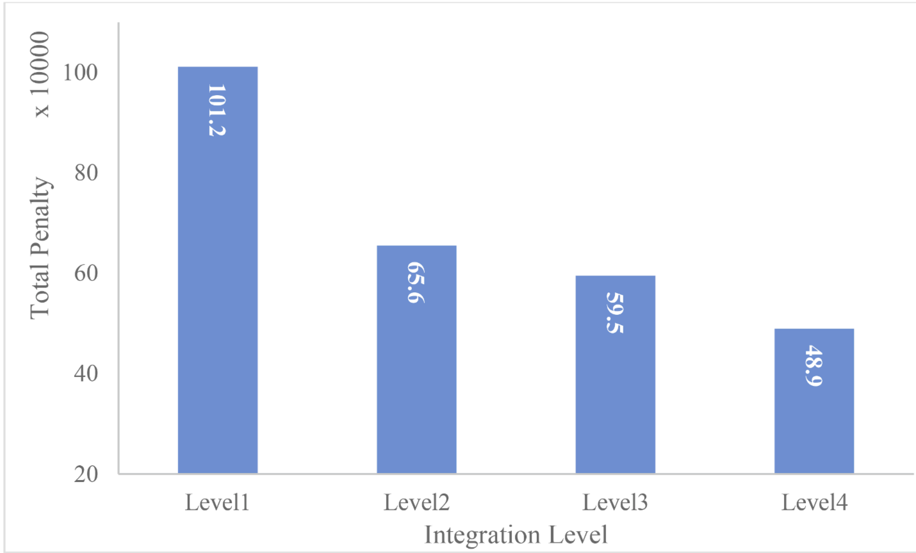

Fig. 11. Comparison of integration levels.

Table 13
Integration level improvements.

Integration Level	Level1	Level2	Level3
Level2	35 %		
Level3	41 %	9 %	
Level4	52 %	24 %	18 %

Table 14
Performance of delivery rules for Level1.

Integration level	Delivery rule	Raw performance	Improvement rate
SIRO-RDM (1011714,0)	Nearest Neighbor [2]	667260,0 tp	34,0%
	Savings Algorithm [3]	638540,0 tp	36,9%
	Sweep Algorithm [4]	700956,0 tp	30,7%
	Hybrid Delivery [6]	700796,0 tp	30,7%
	Priority Algorithm 1 [7]	701772,0 tp	30,6%
	Priority Algorithm 2 [8]	696108,0 tp	31,2%

Table 15
SF results.

Methods	SF1	SF2	SF3	SF4
GA	62748.4 tp	462744.4 tp	1107161.0 tp	1706792.6 tp
ES	53160.3 tp	464487.2 tp	1144144.2 tp	1715942.0 tp
RS	54494.1 tp	470772.0 tp	1133581.0 tp	1611301.0 tp
HS	54346.0 tp	506922.8 tp	1133581.0 tp	1642216.0 tp
SA	54640.1 tp	504247.2 tp	1203021.8 tp	1807164.6 tp

Table 16
SF4 4000 iterations.

Methods	SF4
GA	1607052.2 tp
ES	1715942.0 tp
RS	1611301.0 tp
HS	1595653.0 tp
SA	1665873.4 tp

Table 17
Analysis of DDA rules on the best chromosome.

Rules	SF1	SF2	SF3	SF4
WSLK [0]	62748.4 tp	462744.4 tp	1107161.0 tp	1706792.6 tp
WPPW [1]	331347.6 tp	961279.3 tp	1389143.7 tp	2252548.3 tp
WNOP [2]	318997.8 tp	864993.2 tp	1302128.7 tp	2065005.3 tp
WTWK [3]	283177.8 tp	808833.2 tp	1259172.0 tp	1986143.2 tp

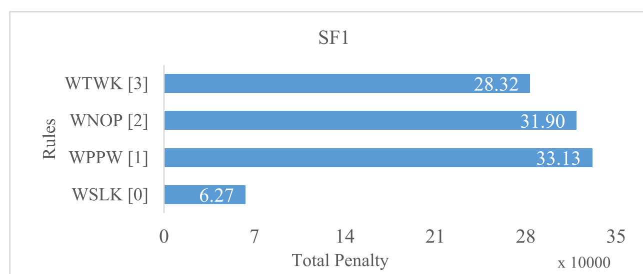

Fig. 12. Performance of DDA rules.

Table 18
Analysis of scheduling rules on the best chromosome.

Rules	SF1	SF2	SF3	SF4
WSPT [0]	62256.4 tp	466706.6 tp	1107161.0 tp	1712032.8 tp
WSOT [1]	62256.4 tp	462744.4 tp	1107161.0 tp	1706792.6 tp
WLOT [2]	63404.4 tp	465614.6 tp	1112453.0 tp	1713536.8 tp
WLPT [3]	63404.4 tp	462744.4 tp	1107161.0 tp	1706792.6 tp
WATC [4]	62256.4 tp	463800.4 tp	1107161.0 tp	1706792.6 tp
ATC [5]	64224.4 tp	463800.4 tp	1107161.0 tp	1706792.6 tp
MS [6]	63404.4 tp	464522.6 tp	1107161.0 tp	1712032.8 tp
WMS [7]	63404.4 tp	464522.6 tp	1107161.0 tp	1712032.8 tp
EDD [8]	62748.4 tp	462744.4 tp	1107161.0 tp	1706792.6 tp
WEDD [9]	63404.4 tp	462744.4 tp	1107161.0 tp	1706792.6 tp

SFs concerning DDA. Within all SFs, the WSLK rule demonstrated superiority as the DDA rule for chromosomes, consistently delivering the best results when utilized with the GA. To illustrate this observation, the outcomes of SF1 are depicted in Fig. 12.

In the case of SF1, like the other SFs, the WSLK rule emerged as the DDA rule that produced the most superior result, and this superiority is

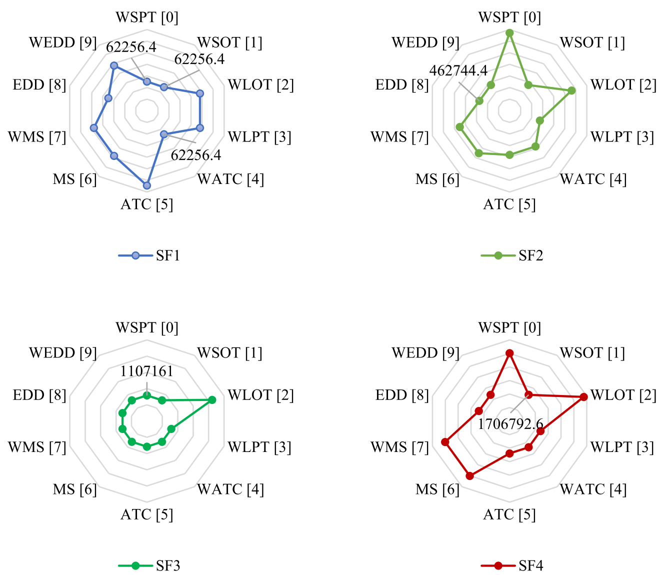

Fig. 13. SF-based scheduling rule performance.

quite significant. The figure shows that the global solution achieved using the WSLK rule outperformed the other rules by almost five, a trend observed across other SFs. However, it is pertinent to note an observed pattern that this differential ratio tends to diminish as the number of jobs increases.

Table 18 presents an analysis of the scheduling rules. In the analysis of the scheduling rules, no rule showed significant dominance. From these results, it is evident that varying rules take precedence in different SFs. Nevertheless, WSOT emerged as the sole rule that delivered the best performance across all four SFs.

Fig. 13 shows a representation of the scheduling rules based on their respective SFs. Considering the objective of this study, which entails minimizing the total penalty score, it is reasonable to infer that the rule positioned closest to the center point of the radar charts exhibits superior performance.

ATC in SF1, WSPT in SF2, and WLOT in SF3 and SF4 had the worst results. The results of the delivery rules for the best chromosomes are presented in Table 19.

The savings algorithm exhibited the best performance among the delivery rules across each SF. Across all simulations, the savings algorithm consistently outperforms the single delivery rule, with results that are 37 % more favorable in SF1, 51 % in SF2, 45 % in SF3, and 49 % in SF4. These percentages highlight that adopting a batch-delivery approach contributes to a superior global solution to this problem. The single, random, and batch delivery rules are significantly underperformed in each SF. A detailed analysis of the delivery rules is presented in Fig. 14.

Across all SFs, the best performance is achieved using the savings algorithm, represented by the rule number [3]. These results emphasize how alterations in the delivery rule of a particular chromosome can greatly impact overall performance. Therefore, the delivery rules have a greater impact on the global solution compared to the scheduling rules. The savings algorithm improved performance by 4.4 % in SF1, 0.3 % in SF2, 7.3 % in SF3, and 5.9 % in SF4 when compared to the next-best rule.

Table 19
Analysis of delivery rules on the best chromosome.

Rules	SF1	SF2	SF3	SF4
Single Delivery [0]	100091,1 tp	946355,8 tp	2010327,6 tp	3364192,0 tp
Batch Delivery [1]	76839,1 tp	605332,4 tp	1530592,0 tp	2268195,6 tp
Nearest Neighbor [2]	65621,7 tp	463915,4 tp	1281968,8 tp	1814371,0 tp
Savings Algorithm [3]	62748,4 tp	462744,4 tp	1107161,0 tp	1706792,6 tp
Sweep Algorithm [4]	72583,2 tp	559516,4 tp	1279739,0 tp	1968475,4 tp
Random Delivery [5]	78584,7 tp	630550,6 tp	1521432,0 tp	2089191,4 tp
Hybrid Delivery [6]	65639,2 tp	526400,4 tp	1194931,0 tp	1898768,8 tp
Priority Algorithm 1 [7]	67762,8 tp	509136,4 tp	1266283,0 tp	2041739,6 tp
Priority Algorithm 2 [8]	67762,8 tp	509136,4 tp	1257995,0 tp	2024923,6 tp

4.1. Comparative statistical analysis of algorithm performances across integration levels

To ensure the reliability of our findings, we conducted a series of statistical tests to validate the observed differences in performance between default and genetic algorithm performance. The comparison between default and GA runs was conducted using an independent two-sample *t*-test, yielding a T-Statistic of 70.37 and an extremely low p-value of 7.30e-42. This p-value indicates a statistically significant difference between the groups, suggesting that the default algorithm and GA exhibit distinct characteristics.

Subsequently, Tukey's Honest Significant Difference test was employed to investigate the disparities among different integration levels (Level1, Level2, Level3, Level4) in terms of GA performance further, as presented in Table 20. The results revealed significant differences between the Level1 and Level2 groups (mean diff = -22780.51, p-adj = 0.0002), as well as between the Level1 and Level4 groups (mean diff = -22511.11, p-adj = 0.0002). However, no significant difference was observed between the Level1 and Level3 groups (p-adj = 0.1594). Furthermore, no significant differences are detected between the Level2 and Level3 groups.

Boxplots are utilized to visually assess the distribution of performance across various integration levels for both algorithmic performances. The analysis reveals discernible variations in performance across different integration levels, as demonstrated in Fig. 15. This graphical depiction facilitates a comprehensive understanding of the diverse performance characteristics exhibited by the algorithms across the evaluated integration levels, thereby providing valuable insights into their comparative efficacy and behavior.

The comparison between the default and GA solutions yielded compelling insights into the efficacy of different integration levels in GA performance. These findings offer valuable insights for researchers and practitioners in optimization and algorithm design.

5. Conclusions

In this study, utilizing various heuristic algorithms, including GA, SA, ES, RS, and HS, provided valuable insights into optimizing manufacturing and supply chain functions. In this approach, each solution is represented by a chromosome. The advantages of integration are demonstrated by examining both the SF results and outcomes of various levels of integration. The evaluation encompasses the performance of the SFs, the effectiveness of the solution methods, and the assessment of each manufacturing and supply chain function by analyzing various rules thoroughly examined in isolation.

Customers requesting incoming jobs to a manufacturer are not always of equal importance. Factors such as the customer's history with the company, frequency of orders, and urgency of work can impact their priority. This study highlighted that customer importance significantly influences due-date assignment, scheduling, and delivery processes. However, in line with just-in-time production philosophy, earliness, tardiness, and promised due date all carry significant weight in the objective function.

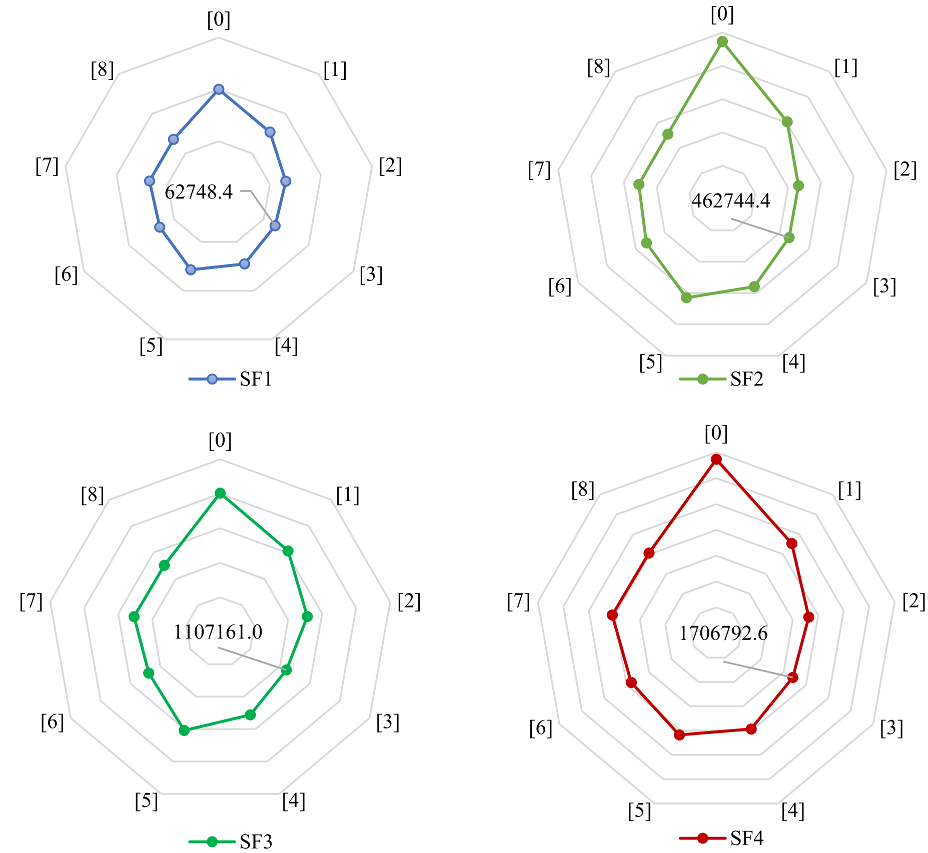

Fig. 14. SF-based delivery rule performance.

Table 20
Tukey's Honestly Significant Difference test results.

group1	group2	mean diff	p-adj	lower	upper	reject
Level1	Level2	−22780.51	0.0002	−36181.83	−9379.19	True
Level1	Level3	−10762.18	0.1594	−24163.50	2639.14	False
Level1	Level4	−22511.11	0.0002	−35912.43	−9109.79	True
Level2	Level3	12018.33	0.0947	−1382.99	25419.65	False
Level2	Level4	269.4	0.9999	−13131.92	13670.72	False
Level3	Level4	−11748.93	0.1064	−25150.25	1652.39	False

This study encompasses four distinct SFs with varying customer numbers and locations. The GA performance for Level4 is 54 % better than the raw performance of Level1, underscoring the evident significance of integration. Furthermore, the global solution improves as the level of integration increases. Even at lower levels, integrating manufacturing and supply chain functions yields more significant benefits than forgoing integration. Notably, ES proved superior on SF1, GA yielded superior results on SF2 and SF3, and the RS algorithm demonstrated greater efficacy on SF4.

Concerning the assignment of due dates, the WSLK rule emerged as the most effective among various rules. Regarding delivery rules, the savings algorithm consistently produced significantly superior results across all SFs. Regarding scheduling rules, no single rule stood out

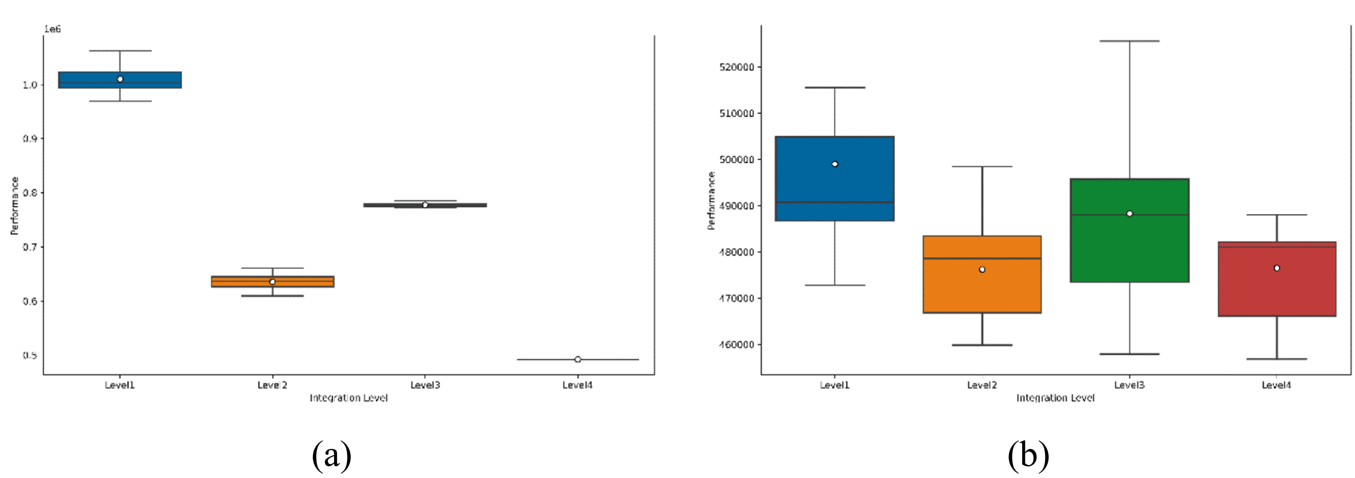

Fig. 15. Distribution of performance across different integration levels for both a) default algorithm and b) genetic algorithm performance.

remarkably, although the WSOT demonstrated greater effectiveness in scheduling performance.

The findings of this study unequivocally manifest the substantial positive impact of integrated communication across the four critical functions in bolstering productivity levels and elevating customer satisfaction within manufacturing settings. The importance of this study lies in its practical demonstration of the various benefits that come from integration all four functions into manufacturing systems. One important aspect of this study is its capability to offer valuable insights into the implications of including a delivery function. The findings of the study have important implications for manufacturing companies, highlighting the opportunity for significant improvements in productivity, profitability, and customer satisfaction. Consequently, companies can significantly fortify their competitive prowess within the global market landscape by integrating all four functions into their production systems.

In this study, utilizing various heuristic algorithms, including GA, SA, ES, RS, and HS, provided valuable insights into optimizing manufacturing and supply chain functions. While these approaches demonstrate significant benefits and are effective for finding good solutions to NP-hard problems within reasonable time frames, they do not guarantee optimality. The solutions obtained are approximate and may vary from the absolute best possible solutions that could be achieved with exhaustive search methods, which are impractical for large-scale problems. As the size and complexity of the problem increase, the computational resources and time required to achieve satisfactory solutions also increase. This scalability issue may limit the applicability of the proposed methods to extremely large or complex problem instances.

In addition to the positive impacts highlighted throughout the study, it is crucial to acknowledge certain limitations that could influence the applicability of the results. The study's reliance on a deterministic approach and the use of a single vehicle in simulations present notable constraints. The deterministic nature might not fully capture the variability and unpredictability inherent in real-world manufacturing environments, potentially limiting the robustness of the proposed solutions under different operational conditions. Furthermore, each manufacturing and supply chain function within this integrated framework holds potential for further exploration through various assumptions or combinations. This study has focused on a set configuration of operations and routing numbers, treating them as homogeneously as possible to maintain consistency.

The primary objective of this study is to demonstrate the feasibility of integrating four distinct manufacturing and supply chain functions to achieve enhanced efficiency. The research findings provide several pertinent suggestions for future inquiry in this field, including the following:

- Expanding process planning information includes the number of jobs and machines on the shop floor, operations executed, and routes linked to each job.
- Stochastic treatment of job arrival times
- Dynamic variables, such as interruptions, urgent jobs, and instances of machine breakdown, are incorporated into the process.
- Inclusion of concepts such as homogeneous or heterogeneous vehicle fleets beyond singular vehicle use for delivery purposes.
- Separate consideration of loading and unloading times in relevant delivery operations.
- Examination of due-window interval dynamics.
- Other terms other than the due date-related costs, such as resource consumption cost in scheduling, transportation cost in delivery, and fixed and variable delivery-related costs, can be examined and included in the problem.
- Energy-efficient production and delivery can be included in the problem. Optimized delivery can reduce the gas emission rate during the delivery phase.
- Additional optimization algorithms such as differential evolution, particle swarm optimization, and artificial bee colony (ABC) can be utilized in future studies.

CRediT authorship contribution statement

Onur Canpolat: Data curation, Formal analysis, Investigation, Methodology, Software, Writing – original draft, Writing – review & editing. **Halil Ibrahim Demir:** Conceptualization, Investigation, Project administration, Resources, Supervision, Writing – original draft, Writing – review & editing. **Caner Erden:** Conceptualization, Formal analysis, Methodology, Resources, Validation, Visualization, Writing – review & editing.

Declaration of Competing Interest

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

Data availability

Data will be made available on request.

References

Arik, O. A. (2023). A heuristic for single machine common due date assignment problem with different earliness/tardiness weights. *OPSEARCH*. https://doi.org/10.1007/s12597-023-00652-1

Arik, O. A., Schutten, M., & Topan, E. (2022). Weighted earliness/tardiness parallel machine scheduling problem with a common due date. *Expert Systems with Applications, 187*, Article 115916. https://doi.org/10.1016/j.eswa.2021.115916

Atsmony, M., & Mosheiov, G. (2024). Common due-date assignment problems with fixed-plus-linear earliness and tardiness costs. *Computers & Industrial Engineering, 188*, Article 109915. https://doi.org/10.1016/j.cie.2024.109915

Ba, L., Li, Y., Yang, M., Wu, X., Liu, Y., Gao, X., & Miao, Z. (2018). A mathematical model for multiworkshop IPPS problem in batch production. *Mathematical Problems in Engineering, 2018*, 7948693. https://doi.org/10.1155/2018/7948693

Berghman, L., Kergosien, Y., & Billaut, J.-C. (2023). A review on integrated scheduling and outbound vehicle routing problems. *European Journal of Operational Research, 311*(1), 1–23. https://doi.org/10.1016/j.ejor.2022.12.036

Bo, V., Bortolini, M., Malaguti, E., Monaci, M., Mora, C., & Paronuzzi, P. (2021). Models and algorithms for integrated production and distribution problems. *Computers & Industrial Engineering, 154*, Article 107003. https://doi.org/10.1016/j.cie.2020.107003

Chen, H.-K., Hsueh, C.-F., & Chang, M.-S. (2009). Production scheduling and vehicle routing with time windows for perishable food products. *Computers & Operations Research, 36*(7), 2311–2319. https://doi.org/10.1016/j.cor.2008.09.010

Chen, J.-S., Liu, H.-S., & Nien, H.-Y. (2007). Minimizing makespan in single machine scheduling with job deliveries to one customer area. *International Journal of Industrial Engineering*, *14*(2), 203–211. Scopus.

Chen, K., Cheng, T. C. E., Huang, H., Ji, M., & Yao, D. (2023). Single-machine scheduling with autonomous and induced learning to minimize total weighted number of tardy jobs. *European Journal of Operational Research, 309*(1), 24–34. https://doi.org/10.1016/j.ejor.2023.01.028

Chen, R.-X., & Li, S.-S. (2020). Minimizing maximum delivery completion time for order scheduling with rejection. *Journal of Combinatorial Optimization, 40*(4), 1044–1064. https://doi.org/10.1007/S10878-020-00649-2

Chen, Z.-L., & Hall, N. G. (2008). Maximum Profit Scheduling. *Manufacturing & Service Operations Management, 10*(1), 84–107. https://doi.org/10.1287/msom.1060.0144

Chu, Y., You, F., Wassick, J. M., & Agarwal, A. (2015). Integrated planning and scheduling under production uncertainties: Bi-level model formulation and hybrid solution method. *Computers & Chemical Engineering, 72*, 255–272. https://doi.org/10.1016/j.compchemeng.2014.02.023

Cura, T. (2008). *Modern Heuristic Techniques and Applications*. Istanbul: Papatya Press. https://www.bkmkitap.com/modern-sezgisel-teknikler-ve-uygulamalari.

Demir, H.İ., Cakar, T., Ipek, M., Uygun, O., & Sari, M. (2015). Process Planning and Due-date Assignment with ATC Dispatching where Earliness, Tardiness and Due-dates are Punished. *Journal of Industrial and Intelligent Information, 3*(3), 197–204. https://doi.org/10.12720/jiii.3.3.197-204.

Demir, H. I., & Erden, C. (2020). Dynamic integrated process planning, scheduling and due-date assignment using ant colony optimization. *Computers & Industrial Engineering, 149*, Article 106799. https://doi.org/10.1016/j.cie.2020.106799

Demir, H. I., Kökçam, A. H., & Erden, C. (2023). In *Integrated Process planning, scheduling, and due-date assignment*. CRC Press. https://doi.org/10.1201/9781003215295.

Demir, H. İ., & Phanden, R. K. (2019). Due-Date Agreement in Integrated Process Planning and Scheduling Environment Using Common Meta-Heuristics. In *Integration* *of Process Planning and Scheduling* (pp. 161–184). CRC Press. https://doi.org/10.1201/9780429021305-8.

Demir, H. I., Kökçam, A. H., & Erden, C. (2024). *Integrated Process planning, Scheduling, and Due-date Assignment*. CRC Press. https://doi.org/10.1201/9781003215295

Demir, H.İ., Phanden, R., Kökçam, A., Erkayman, B., & Erden, C. (2021). Hybrid evolutionary strategy and simulated annealing algorithms for integrated process planning, scheduling and due-date assignment problem. *Academic Platform Journal of Engineering and Science, 9*(1), 86–91. https://doi.org/10.21541/apjes.764150.

Demir, H. İ., & Taskin, H. (2005). *Integrated Process Planning, Scheduling and Due-Date Assignment* [PhD Thesis]. Sakarya University.

Erden, C. (2019). *Dynamic Integrated Process Planning, Scheduling and Due Date Assignment* [PhD Thesis].

Erden, C., Demir, H. I., & Canpolat, O. (2023). A modified integer and categorical PSO algorithm for solving integrated process planning, dynamic scheduling, and due date assignment problem. *Scientia Iranica, 30*(2), 738–756. https://doi.org/10.24200/sci.2021.55250.4130.

Erden, C., Demir, H.İ., & Kökçam, A. H. (2019). Solving Integrated Process Planning, Dynamic Scheduling, and Due Date Assignment Using Metaheuristic Algorithms. *Mathematical Problems in Engineering, 2019*, 1572614. https://doi.org/10.1155/2019/1572614

Fu, L.-L., Aloulou, M. A., & Triki, C. (2017). Integrated production scheduling and vehicle routing problem with job splitting and delivery time windows. *International Journal of Production Research, 55*(20), 5942–5957. https://doi.org/10.1080/00207543.2017.1308572

Garcia, J. M., & Lozano, S. (2004). Production and vehicle scheduling for ready-mix operations. *Computers & Industrial Engineering, 46*(4), 803–816. https://doi.org/10.1016/j.cie.2004.05.011

Ghannadpour, S. F., & Zarrabi, A. (2018). Multi-objective heterogeneous vehicle routing and scheduling problem with energy minimizing. *Swarm and Evolutionary Computation*. https://doi.org/10.1016/j.swevo.2018.08.012

Göçken, T., Yaktubay, M., & Kılıç, F. (2018). A multi objective genetic algorithm approach for the solution of vehicle routing problem with time windows. *Gazi University Journal of Science Part C: Design and Technology*. https://doi.org/10.29109/gujsc.397543.

Gordon, V., Proth, J. M., & Chu, C. (2002). A survey of the state-of-the-art of common due date assignment and scheduling research. *European Journal of Operational Research, 139*(1), 1–25. https://doi.org/10.1016/S0377-2217(01)00181-3

Grigoreva, N. (2020). Multiprocessor scheduling problem with release and delivery times. *Annals of Computer Science and Information Systems, 21*, 263–269. https://annals-csis.org/proceedings/2020/drp/33.html.

Guo, Y. W., Li, W. D., Mileham, A. R., & Owen, G. W. (2009). Applications of particle swarm optimisation in integrated process planning and scheduling. *Robotics and Computer-Integrated Manufacturing, 25*(2), 280–288. https://doi.org/10.1016/j.rcim.2007.12.002

Han, D., Yang, Y., Wang, D., Cheng, T. C. E., & Yin, Y. (2019). Integrated production, inventory, and outbound distribution operations with fixed departure times in a three-stage supply chain. *Transportation Research Part E: Logistics and Transportation Review, 125*, 334–347. https://doi.org/10.1016/j.tre.2019.03.014

Hou, Y., Fu, Y., Gao, K., Zhang, H., & Sadollah, A. (2022). Modelling and optimization of integrated distributed flow shop scheduling and distribution problems with time windows. *Expert Systems with Applications, 187*, Article 115827. https://doi.org/10.1016/j.eswa.2021.115827

Huang, M., Du, B., & Guo, J. (2023). A hybrid collaborative framework for integrated production scheduling and vehicle routing problem with batch manufacturing and soft time windows. *Computers & Operations Research, 159*, Article 106346. https://doi.org/10.1016/j.cor.2023.106346

Ismail, M., Sale, M., Yu, Y., Pillai, N., Liu, S., Pflug, B., & Bies, R. (2022). Development of a genetic algorithm and NONMEM workbench for automating and improving population pharmacokinetic/pharmacodynamic model selection. *Journal of Pharmacokinetics and Pharmacodynamics, 49*(2), 243–256. https://doi.org/10.1007/s10928-021-09782-9

Janiak, A., Janiak, W. A., Krysiak, T., & Kwiatkowski, T. (2015). A survey on scheduling problems with due windows. *European Journal of Operational Research, 242*(2), 347–357. https://doi.org/10.1016/j.ejor.2014.09.043

Jin, L., Zhang, C., Shao, X., Yang, X., & Tian, G. (2016). A multi-objective memetic algorithm for integrated process planning and scheduling. *The International Journal of Advanced Manufacturing Technology, 85*(5), 1513–1528. https://doi.org/10.1007/s00170-015-8037-7

Karaoğlan, İ., & Kesen, S. E. (2017). The coordinated production and transportation scheduling problem with a time-sensitive product: A branch-and-cut algorithm. *International Journal of Production Research, 55*(2), 536–557. https://doi.org/10.1080/00207543.2016.1213916

Keskinocak, P., & Tayur, S. (2004). Due Date Management Policies. In D. Simchi-Levi, S. D. Wu, & Z.-J. Shen (Eds.), *Handbook of Quantitative Supply Chain Analysis* (Vol. 74, pp. 485–554). Springer US. https://doi.org/10.1007/978-1-4020-7953-5_12.

Koulamas, C., & Kyparisis, G. J. (2022). Flow shop scheduling with two distinct job due dates. *Computers & Industrial Engineering, 163*, Article 107835. https://doi.org/10.1016/j.cie.2021.107835

Lacomme, P., Moukrim, A., Quilliot, A., & Vinot, M. (2016). The Integrated Production and Transportation Scheduling Problem based on a GRASP×ELS resolution scheme. *IFAC-PapersOnLine, 49*(12), 1466–1471. https://doi.org/10.1016/j.ifacol.2016.07.778

Lee, H., & Kim, S.-S. (2001). Integration of process planning and scheduling using simulation based genetic algorithms. *The International Journal of Advanced Manufacturing Technology, 18*(8), 586–590. https://doi.org/10.1007/s001700170035

Leung, C. W., Wong, T. N., Mak, K. L., & Fung, R. Y. K. (2010). Integrated process planning and scheduling by an agent-based ant colony optimization. *Computers & Industrial Engineering, 59*(1), 166–180. https://doi.org/10.1016/j.cie.2009.09.003

Li, F., Chen, Z.-L., & Tang, L. (2017). Integrated production, inventory and delivery problems: complexity and algorithms. *INFORMS Journal on Computing, 29*(2), 232–250. https://doi.org/10.1287/ijoc.2016.0726

Li, X., & Gao, L. (2020). *Effective Methods for Integrated Process Planning and Scheduling* (Vol. 2). https://doi.org/10.1007/978-3-662-55305-3

Lin, C. T., & Jou, C. P. (2000). GA-based fuzzy reinforcement learning for control of a magnetic bearing system. *IEEE Transactions on Systems, Man, and Cybernetics. Part B, Cybernetics, 30*(2), 276–289. https://doi.org/10.1109/3477.836376

Liu, H., Guo, Z., & Zhang, Z. (2021). A hybrid multi-level optimisation framework for integrated production scheduling and vehicle routing with flexible departure time. *International Journal of Production Research, 59*(21), 6615–6632. https://doi.org/10.1080/00207543.2020.1821927

Liu, Q., Li, X., Gao, L., & Fan, J. (2022). A multi-MILP model collaborative optimization method for integrated process planning and scheduling problem. *IEEE Transactions on Engineering Management*. https://ieeexplore.ieee.org/abstract/document/9910956/?casa_token=Rsjf9qtpGdsAAAAA:DeHantBYGb-Haa_jK9drRzn78h6Mju-_5gWwSuh2m_JUlmrDIDPQ-c3io8j0eKD4Dxdn6w8Za0.

Long, J., Pardalos, P. M., & Li, C. (2022). Level-based multi-objective particle swarm optimizer for integrated production scheduling and vehicle routing decision with inventory holding, delivery, and tardiness costs. *International Journal of Production Research, 60*(11), 3319–3338. https://doi.org/10.1080/00207543.2021.1919780

Luo, G., Wen, X., Li, H., Ming, W., & Xie, G. (2017). An effective multi-objective genetic algorithm based on immune principle and external archive for multi-objective integrated process planning and scheduling. *The International Journal of Advanced Manufacturing Technology, 91*(9), 3145–3158. https://doi.org/10.1007/s00170-017-0020-z

Luo, Q., Fan, Q., Deng, Q., Guo, X., Gong, G., & Liu, X. (2023). Solving bi-objective integrated scheduling problem of production, inventory and distribution using a modified NSGA-II. *Expert Systems with Applications, 225*, Article 120074. https://doi.org/10.1016/j.eswa.2023.120074

Márquez, C. R. H., & Ribeiro, C. C. (2022). Shop scheduling in manufacturing environments: A review. *International Transactions in Operational Research, 29*(6), 3237–3293. https://doi.org/10.1111/itor.13108

Mohapatra, P., Nayak, A., Kumar, S. K., & Tiwari, M. K. (2015). Multi-objective process planning and scheduling using controlled elitist non-dominated sorting genetic algorithm. *International Journal of Production Research, 53*(6), 1712–1735. https://doi.org/10.1080/00207543.2014.957872

Moon, C., Lee, Y. H., Jeong, C. S., & Yun, Y. (2008). Integrated process planning and scheduling in a supply chain. *Computers & Industrial Engineering, 54*(4), 1048–1061. https://doi.org/10.1016/j.cie.2007.06.018

Moons, S., Ramaekers, K., Caris, A., & Arda, Y. (2017). Integrating production scheduling and vehicle routing decisions at the operational decision level: A review and discussion. *Computers & Industrial Engineering, 104*, 224–245. https://doi.org/10.1016/j.cie.2016.12.010

Mor, B., Shabtay, D., & Yedidsion, L. (2021). Heuristic algorithms for solving a set of NP-hard single-machine scheduling problems with resource-dependent processing times. *Computers & Industrial Engineering, 153*, Article 107024. https://doi.org/10.1016/j.cie.2020.107024

Morad, N., & Zalzala, A. (1999). Genetic algorithms in integrated process planning and scheduling. *Journal of Intelligent Manufacturing, 10*(2), 169–179. https://doi.org/10.1023/A:1008976720878

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. https://doi.org/10.1016/j.apm.2009.09.002

Pan, L., Sun, X., Wang, J.-B., Zhang, L.-H., & Lv, D.-Y. (2023). Due date assignment single-machine scheduling with delivery times, position-dependent weights and deteriorating jobs. *Journal of Combinatorial Optimization, 45*(4), 100. https://doi.org/10.1007/s10878-023-01025-6

Petrović, M., Mitić, M., Vuković, N., & Miljković, Z. (2016). Chaotic particle swarm optimization algorithm for flexible process planning. *The International Journal of Advanced Manufacturing Technology, 85*(9), 2535–2555. https://doi.org/10.1007/s00170-015-7991-4

Petrović, M., Petronijević, J., Mitić, M., Vuković, N., Miljković, Z., & Babić, B. (2016). The ant lion optimization algorithm for integrated process planning and scheduling. *Applied Mechanics and Materials, 834*, 187–192. https://doi.org/10.4028/www.scientific.net/AMM.834.187

Phanden, R. K., Jain, A., & Davim, J. P. (Eds.). (2019). *Integration of Process Planning and Scheduling: Approaches and Algorithms* (1st edition). CRC Press.

Phanden, R. K., Jain, A., & Verma, R. (2011). Integration of process planning and scheduling: A state-of-the-art review. *International Journal of Computer Integrated Manufacturing, 24*(6), 517–534. https://doi.org/10.1080/0951192X.2011.562543

Qian, J., & Han, H. (2022). The due date assignment scheduling problem with the deteriorating jobs and delivery time. *Journal of Applied Mathematics and Computing, 68*(4), 2173–2186. https://doi.org/10.1007/s12190-021-01607-9

Reiter, B. S., Makuschewitz, T., Novaes, A. G. N., Frazzon, E. M., & Lima, Jr. (2011). An approach for the sustainable integration of production and transportation scheduling. *International Journal of Logistics Systems and Management, 10*(2), 158–179. Scopus. https://doi.org/10.1504/IJLSM.2011.042626.

Scallan, P. (2003). 2—What is process planning? In P. Scallan (Ed.), *Process Planning* (pp. 35–62). Butterworth-Heinemann. https://doi.org/10.1016/B978-075065129-5/50003-9.

Shabtay, D., Mosheiov, G., & Oron, D. (2022). Single machine scheduling with common assignable due date/due window to minimize total weighted early and late work.

European Journal of Operational Research, 303(1), 66–77. https://doi.org/10.1016/j.ejor.2022.02.017

Shahin Moghadam, S., Fatemi Ghomi, S. M. T., & Karimi, B. (2014). Vehicle routing scheduling problem with cross docking and split deliveries. *Computers & Chemical Engineering, 69*, 98–107. https://doi.org/10.1016/j.compchemeng.2014.06.015

Shen, X.-N., & Yao, X. (2015). Mathematical modeling and multi-objective evolutionary algorithms applied to dynamic flexible job shop scheduling problems. *Information Sciences, 298*, 198–224. https://doi.org/10.1016/j.ins.2014.11.036

Sobeyko, O., & Mönch, L. (2017). Integrated process planning and scheduling for large-scale flexible job shops using metaheuristics. *International Journal of Production Research, 55*(2), 392–409. https://doi.org/10.1080/00207543.2016.1182227

Solina, V., & Mirabelli, G. (2021). Integrated production-distribution scheduling with energy considerations for efficient food supply chains. *Procedia Computer Science, 180*, 797–806. https://doi.org/10.1016/j.procs.2021.01.355

Taguchi, G. (1986). *Introduction to Quality Engineering: Designing Quality into Products and Processes*. Quality Resources.

Tan, W., & Khoshnevis, B. (2000). Integration of process planning and scheduling—A review. *Journal of Intelligent Manufacturing, 11*(1), 51–63. https://doi.org/10.1023/A:1008952024606

Tarhan, İ., & Oğuz, C. (2021). Generalized order acceptance and scheduling problem with batch delivery: Models and metaheuristics. *Computers & Operations Research, 134*, Article 105414. https://doi.org/10.1016/j.cor.2021.105414

Tarhini, A., Danach, K., & Harfouche, A. (2022). Swarm intelligence-based hyper-heuristic for the vehicle routing problem with prioritized customers. *Annals of Operations Research*, 1–22.

Tonizza Pereira, M., & Seido Nagano, M. (2022). Hybrid metaheuristics for the integrated and detailed scheduling of production and delivery operations in no-wait flow shop systems. *Computers & Industrial Engineering, 170*, Article 108255. https://doi.org/10.1016/j.cie.2022.108255

Ullrich, C. A. (2013). Integrated machine scheduling and vehicle routing with time windows. *European Journal of Operational Research, 227*(1), 152–165. https://doi.org/10.1016/j.ejor.2012.11.049

Vinod, V., & Sridharan, R. (2011). Simulation modeling and analysis of due-date assignment methods and scheduling decision rules in a dynamic job shop production system. *International Journal of Production Economics, 129*(1), 127–146. https://doi.org/10.1016/j.ijpe.2010.08.017

Wang, D., & Li, Z. (2019). Bicriterion scheduling with a negotiable common due window and resource-dependent processing times. *Information Sciences, 478*, 258–274. https://doi.org/10.1016/j.ins.2018.11.023

Wang, J., Fan, X., Zhang, C., & Wan, S. (2014). A graph-based ant colony optimization approach for integrated process planning and scheduling. *Chinese Journal of Chemical Engineering, 22*(7), 748–753. https://doi.org/10.1016/j.cjche.2014.05.011

Wang, S., Wu, R., Chu, F., & Yu, J. (2020). Variable neighborhood search-based methods for integrated hybrid flow shop scheduling with distribution. *Soft Computing, 24*(12), 8917–8936. https://doi.org/10.1007/s00500-019-04420-6

Xu, X., Yin, G., & Wang, C. (2021). Multitasking scheduling with batch distribution and due date assignment. *Complex & Intelligent Systems, 7*(1), 191–202. https://doi.org/10.1007/s40747-020-00184-x

Yağmur, E., & Kesen, S. E. (2021). Multi-trip heterogeneous vehicle routing problem coordinated with production scheduling: Memetic algorithm and simulated annealing approaches. *Computers & Industrial Engineering, 161*, Article 107649. https://doi.org/10.1016/j.cie.2021.107649

Yağmur, E., & Kesen, S. E. (2023). Bi-objective coordinated production and transportation scheduling problem with sustainability: Formulation and solution approaches. *International Journal of Production Research, 61*(3), 774–795. https://doi.org/10.1080/00207543.2021.2017054

Yang, F., Davari, M., Wei, W., Hermans, B., & Leus, R. (2022). Scheduling a single parallel-batching machine with non-identical job sizes and incompatible job families. *European Journal of Operational Research, 303*(2), 602–615. https://doi.org/10.1016/j.ejor.2022.03.027

Yin, Y., Li, D., Wang, D., & Cheng, T. C. E. (2021). Single-machine serial-batch delivery scheduling with two competing agents and due date assignment. *Annals of Operations Research, 298*(1), 497–523. https://doi.org/10.1007/s10479-018-2839-6

Yu, M. R., Yang, B., & Chen, Y. (2018). Dynamic integration of process planning and scheduling using a discrete particle swarm optimization algorithm. *Advances in Production Engineering & Management, 13*(3), 279–296.

Yu, M., Zhang, Y., Chen, K., & Zhang, D. (2015). Integration of process planning and scheduling using a hybrid GA/PSO algorithm. *The International Journal of Advanced Manufacturing Technology, 78*(1), 583–592. https://doi.org/10.1007/s00170-014-6669-7

Yue, Q., & Zhou, S. (2021). Due-window assignment scheduling problem with stochastic processing times. *European Journal of Operational Research, 290*(2), 453–468. https://doi.org/10.1016/j.ejor.2020.08.029

Zhan, Y., & Wan, G. (2018). Vehicle routing and appointment scheduling with team assignment for home services. *Computers & Operations Research, 100*, 1–11. https://doi.org/10.1016/j.cor.2018.07.006

Zhang, C., Li, Y., Cao, J., Yang, Z., & Coelho, L. C. (2022). Exact and matheuristic methods for the parallel machine scheduling and location problem with delivery time and due date. *Computers & Operations Research, 147*, Article 105936. https://doi.org/10.1016/j.cor.2022.105936

Zhang, L., & Wong, T. N. (2015). An object-coding genetic algorithm for integrated process planning and scheduling. *European Journal of Operational Research, 244*(2), 434–444. https://doi.org/10.1016/j.ejor.2015.01.032

Zhang, R., & Wu, C. (2012). A double-layered optimisation approach for the integrated due date assignment and scheduling problem. *International Journal of Production Research, 50*(1), 5–22. https://doi.org/10.1080/00207543.2011.571440

Zhang, S., & Wong, T. N. (2018). Integrated process planning and scheduling: An enhanced ant colony optimization heuristic with parameter tuning. *Journal of Intelligent Manufacturing, 29*(3), 585–601. https://doi.org/10.1007/s10845-014-1023-3

Zhao, C.-L., & Tang, H.-Y. (2014). A note on single machine scheduling and due date assignment with general position-dependent processing times. *International Journal of Production Research, 52*(9), 2807–2814. https://doi.org/10.1080/00207543.2014.886025

Zhao, J., Wang, T., Pedrycz, W., & Wang, W. (2021). Granular prediction and dynamic scheduling based on adaptive dynamic programming for the blast furnace gas system. *IEEE Transactions on Cybernetics, 51*(4), 2201–2214. https://doi.org/10.1109/TCYB.2019.2901268