Hyper-heuristic Ant Colony Optimization for solving the integrated distributed permutation flow shop problem and multiple compartments vehicle routing problem with simultaneous deterministic delivery and fuzzy pickup

Yan Geng, Bin Qian(✉), Ning Guo and Rong Hu

School of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China
bin.qian@vip.163.com

Abstract. Production and transportation are two essential activities in supply chain management, decision-makers strive to enhance the operational efficiency of these two stages to maximize business interests. In this paper, we consider the integrated distributed permutation flow shop problem (IDFSP) and multiple compartments vehicle routing problem with simultaneous deterministic delivery and fuzzy pickup (IDFSP_MCVRPSDDFP). The IDFSP_MCVRPSDDFP aims to simultaneously minimize cost and carbon emissions caused by both production and transporation. To address the IDFSP_MCVRPSDDFP, we propose a hyper-heuristic ant colony optimization algorithm (HH_ACO). The HH_ACO is composed of two main components: a hyper-heuristic algorithm (HHA) and an ant colony optimization algorithm (ACO). To enhance the efficiency of local search, we design six heuristic operations within the low-level heuristics (LLHs). Meanwhile, the ACO is utilized to enhance the performance of the high-level heuristics (HLS) within the HHA. Experimental simulations and data analysis have validated that HH_ACO can effectively solve IDFSP_MCVRPSDDFP.

Keywords: distributed permutation flow shop; multi-objective optimization; hyper heuristic algorithm.

1 Introduction

Production and transportation play crucial roles in the supply chain system. Studies have shown that by effectively addressing the integrated production and transportation scheduling problem (IPTSP), enterprises can reduce operational costs by 3% to 20%. Therefore, for enterprises to maximize overall benefits, it is necessary to optimize the IPTSP.

Currently, scholars have conducted relevant research on the IPTSP. The investigation can be classified into two groups. The first type is the integrated scheduling problem of single-factory production and transportation. Moons et al. [1] conducted a literature review on integrated scheduling problems. From this article, it can be seen that there have been many studies on the integration problem of single factory production and transportation in existing literature. For example, Karaoglan et al. [2] proposed a branch-and-price algorithm to solve the integrated scheduling problem of single-factory production and transportation, with the objective of minimizing the makespan.Guo et al. [3] addressed the integrated scheduling problem with the objective of minimizing the total cost. They designed a hybrid two-layer algorithm that combines the memetic algorithm to solve this problem. The proposed algorithm's effectiveness was validated through experimental simulations. Zou et al. [4] utilized an improved genetic algorithm to address the integrated scheduling problem of single-machine production and transportation, aiming to minimize order delivery time. The second type is the integrated scheduling problem of distributed production and transportation.Yang et al. [5] used VND algorithm and iterative greedy algorithm based on product destruction to solve the integrated scheduling problem of distributed flexible assembly lines and transportation, with the optimization goal of minimizing the total cost of distribution and delay.

Although scholars have conducted relevant research on the IPTSP, existing literature has not taken into account other situations that exist in real vehicle transportation; For example, vehicles need to complete simultaneous pick-up and delivery tasks at the customer's location, where customer demands are diverse and demand is fuzzy. In summary, this article extends the traditional integrated distributed permutation flow-shop problem and transportation problem. During the production phase, the study addresses distributed production in factories, concentrating on the allocation of jobs to factories and the determination of the processing sequence for each job within the respective factory. In the transportation phase, simultaneous pick-up and delivery of customer demands are considered, introducing fuzzy requirements for pick-up tasks. Additionally, this study takes into account the characteristic of customers having diverse demands. Therefore the integrated distributed permutation flow shop problem (DFSP) and multiple compartments vehicle routing problem with simultaneous deterministic delivery and fuzzy pickup (IDFSP_MCVRPSDDFP) studied in this article has profound theoretical value and practical significance.

In terms of solving, IDFSP_MCVRPSDDFP as a complex combinatorial optimization problem is NP-hard, because it can be reduced to an NP-hard problem DFSP [6]. Employing intelligent algorithms allows obtaining satisfactory solutions within a relatively short time, thereby providing decision-makers with viable options. The hyper-heuristic algorithm (HHA) is a novel type of intelligent optimization algorithm characterized by a two-layer structure. It comprises a high-level strategy (HLS) that manipulates or manages low-level heuristics (LLHs). During the iterative process of HHA, the HLS dynamically controls LLHs to continuously combine into new heuristic sequences. This enables exploration of different regions within the problem solution space. The Ant Colony Optimization (ACO) algorithm was initially introduced by the Italian scholar Marco Dorigo in the 1990s [7], primarily for solving the Traveling Salesman Problem (TSP). Through a positive feedback mechanism, the algorithm can achieve satisfactory results in a short time.Based on the superior performance of HHA and ACO algorithms, this paper proposes a hyper-heuristic ant colony optimization (HH_ACO) algorithm to solve IDFSP_MCVRPSDDFP.

The remainder of this study is as bellow: The problem description and mathematical modeling are described in Section 2. The proposed algorithm is introduced in Section 3. Next, the experimental results are discussed in Section 4. In the end, Section 5 gives the conclusions of this study.

2 Problem Description and Mathematical Model

2.1 Problem Description

The IDFSP_MCVRPSDDFP problem studied in this paper can be described as follows: allocating N jobs to F factories with identical configurations but located in different positions for processing. After the completion of job processing, they are transported to customers by the vehicles of each factory. The entire process is divided into production and transportation stages. In the production stage, each factory has the same configuration, including M machines and multiple vehicles. All jobs can be assigned for processing in any factory, but once a job is assigned to a specific factory, it cannot be assigned to any other factory. Jobs need to be processed on each machine in the same order. At the same time, each machine can process only one job, and different jobs are independent of each other. In the transportation stage: after all products have been processed, the vehicles from each factory transport the products to customers. Each vehicle has multiple compartments to store different types of products. While serving customers, each customer not only has delivery tasks but also needs to complete pickup tasks at their location. After completing all the delivery and pickup tasks for customers, the vehicle returns to the factory from which it originally departed, awaiting the next transportation task.

In real-life situations, the delivery demand of customers is known before the vehicles arrive, but the pickup demand is typically uncertain. Customers often provide an approximate range for their pickup demand based on experience. For example, a customer's pickup demand may be in the range of 2kg to 4kg, with the most likely value being 3kg. The actual pickup demand can only be determined when the vehicle arrives at the customer's location.Therefore, this paper introduces a fuzzy variable $\tilde{q}_{ip} = \left(q_{1,ip}, q_{2,ip}, q_{3,ip}\right)$ to represent the fuzzy pickup demand for the p product of customer i, where $q_{1,ip}, q_{2,ip}, q_{3,ip}$ denotes the lower bound, most likely value, and upper bound of a triangular fuzzy number, respectively.

Due to the operation of machines in the production stage and the travel of vehicles in the transportation stage, corresponding costs and carbon emissions are generated. Therefore, the total cost should be equal to the sum of the costs generated in the production and transportation phases. Similarly, the total carbon emissions should be equal to the sum of the carbon emissions generated by factory machines and vehicle travel. Thus, this paper aims to provide decision-makers with a set of solutions that balance both costs and carbon emissions.

At the same time, this paper satisfies the following assumptions:

(1) Each job can only be assigned to one factory.
(2) Each factory is assigned at least one job.
(3) Each machine can process only one job at the same time.
(4) The machines operate continuously without interruptions.
(5) The vehicles satisfy fuzzy chance constraints during the process of serving customers.

2.2 Mathematical Model

The mathematics involved in this paper is shown in **Table 1** .

Table 1. Symbol definition and description of the IDFSP_MCVRPSDDFP

Indexes	***Description***
i, j	Index of job (or customer), $i, j \in \{1, 2, \cdots, N\}$
m	Index of machine, $m \in \{1, 2, \cdots, M\}$
f	Index of factory, $f \in \{1, 2, \cdots, F\}$
k	Index of vehicle, $k \in \{1, 2, \cdots, K\}$
p	Index of job category (or vehicle compartment), $p \in \{1, 2, \cdots, P\}$
Parameters	***Description***
$p_{i,m}$	The processing time of job *i* on machine *m*.
$C_{i,m}$	Makespan of job *i* on machine *m*.
C_f	Makespan of factory *f*.
$\tilde{q}_{ip}$	The fuzzy collection quantity for product type p of customer i
d_{ip}	The determined delivery quantity for product type *p* of customer *i*
Q_p^k	Maximum load capacity of compartment *p* for vehicle *k*
Q_{ip}^k	The load capacity of compartment *p* of vehicle *k* when leaving customer *i*
$\Delta\tilde{Q}_{ip}^k$	The fuzzy remaining loading capacity of compartment *p* when vehicle *k* departs from customer *i* .
W	Vehicle weight
w_i	Load capacity of the vehicle after leaving customer i
d_{ij}	Distance from customer i to customer j
v_{ij}	The speed of the vehicle from customer i to customer j
Set	***Description***
π	Total sequence of jobs $\pi = \{\pi(l) \mid l = 1, 2, \cdots, N\}$
π_f	The sequence of jobs in factory f , $\pi_f = \{\pi_f(l) \mid l = 1, 2, \cdots, n_f\}$
π_f^k	The sequence of jobs carried by vehicle k from factory f , $\pi_f^k = \{\pi_f^k(a) \mid a = 1, 2, \cdots, n_f^k\}$
$\pi_f^{'k}$	The sequence of customers served by vehicle k from factory f, $\pi_f^{'k} = \{\pi_f^{'k}(b) \mid b = 1, 2, \cdots, n_f^k\}$
Variables	***Description***
x_{ijk}	The variable is 1 when vehicle k travels from point i to j, otherwise 0

The mathematical model established based on the above description is as follows: the completion time of job $\pi_f(l)$ in the production stage is calculated as follows:

$$C_{\pi_f(1),1} = p_{\pi_f(1),1}, \forall f \in \{1,2,\cdots,F\} \tag{1}$$

$$C_{\pi_f(l),1} = C_{\pi_f(l-1),1} + p_{\pi_f(l),1}, \forall f \in \{1,2,\cdots,F\}, l \in \{2,3,\cdots n_f\} \tag{2}$$

$$C_{\pi_f(1),i} = C_{\pi_f(1),(m-1)} + p_{\pi_f(1),m}, \forall f \in \{1,2,\cdots,F\}, m \in \{2,3,\cdots M\} \tag{3}$$

$$\begin{aligned} &C_{\pi_f(l),m} = \max\left\{C_{\pi_f(l-1),m}, C_{\pi_f(l),(m-1)}\right\} + p_{\pi_f(l),m}, \\ &\forall f \in \{1,2,\cdots,F\}, m \in \{2,3,\cdots M\}, l \in \{2,3,\cdots n_f\} \end{aligned} \tag{4}$$

$$C_f = C_{(\pi_f(n_f)),M}, \forall f \in \{1,2,\cdots F\} \tag{5}$$

Therefore, the production cost TPC and the carbon emissions TEC during production phase are calculated as follows:

$$TPC = \alpha \times \max\{C_f\}, \forall f \in \{1,2,\cdots,F\} \tag{6}$$

$$PEC_f = C_1 \times \sum_{m=1}^{M} \sum_{i=1}^{n_f} p_{i,m} \forall f \in \{1,2,\cdots,F\} \tag{7}$$

$$SEC_f = C_2 \times \left(M \times C_f - \sum_{m=1}^{M} \sum_{i=1}^{n_f} p_{i,m} \right) \forall f \in \{1,2,\cdots,F\} \tag{8}$$

$$TEC = \sum_{f=1}^{F} \left(PEC_f + SEC_f \right) \tag{9}$$

In the above equations, PEC_f and SEC_f represent the energy consumption during machine processing and idle state, respectively. C_1 and C_2 denote the carbon emission coefficients per unit of processing time and idle time for the machine, respectively. In this paper, they are set as 1.1 and 0.7, respectively.

In the transportation stage, due to the fuzzy nature of customer pickup demands, capacity constraints are more complex compared to constraints under deterministic demands. Therefore, this paper constructs a fuzzy chance-constrained programming model based on fuzzy credibility theory to address the fuzzy pickup demands of customers. Therefore, the models for vehicle load constraints and fuzzy chance constraints during the transportation stage are as follows.

Assuming a certain vehicle k has already determined the sequence of customers to be served as $\{s,t,u\}$, the decision variable x_{ujk} influences whether the vehicle can proceed to the next customer point j after serving the customers in the sequence u. When $x_{ujk} = 1$, the vehicle k proceeds to serve customer j after serving customer u, otherwise, it returns to the factory. The specific equation is as follows :

$$x_{ujk} = \begin{cases} 1, C_r\left(\tilde{q}_{jp} \le \Delta\tilde{Q}_{up}^k\right) \ge \alpha \\ 0, C_r(\tilde{q}_{jp} > \Delta\tilde{Q}_{up}^k) < \alpha \end{cases} \tag{10}$$

In Equation (10), $C_r\left(\tilde{q}_{jp} \le \Delta\tilde{Q}_{upk}\right)$ represents a feasibility measure $\tilde{q}_{jp} \le \Delta\tilde{Q}_{upk}$ of a triangular fuzzy number, and its calculation method refers to the literature [8] where $C_r \in [0,1]$, and a larger C_r indicates a greater chance for the vehicle to serve the next customer j. To design the capacity constraint under fuzzy demand, we introduce the decision-maker's preference value α. If $C_r\left(\tilde{q}_j \le \Delta\tilde{Q}_{up}^k\right) \ge \alpha$, the vehicle continues to serve the next customer point j and continues to assess the remaining unserved customers. Otherwise, the vehicle returns to the factory, and another vehicle is dispatched to assess the unserved customers. The preference value α reflects the decision-maker's risk attitude. A smaller α indicates that the decision-maker is more inclined to take the risk of potential failures to fully utilize the remaining space of the vehicle. Conversely, a larger α indicates that the decision-maker would rather dispatch an additional vehicle to continue serving the remaining customers than take the risk of failure.

At the same time, to more accurately reflect the relationship between the fuzzy pickup demand of the next customer point j and the remaining space in the compartment, the preference value α in Equation (10) is set to an adaptive value as specifically shown in Equation (11). To determine the change in compartment capacity during vehicle transportation, the expected value $E(\bullet)$ of fuzzy numbers is introduced, calculated as shown in equation (12).

$$\alpha_j = \frac{E\left(\tilde{Q}_{up}^k\right) - d_{jp}}{Q_p^k} \tag{11}$$

$$E(\tilde{q}_i) = \frac{q_{1,i} + 2q_{2,i} + q_{3,i}}{4} \tag{12}$$

Therefore, the transportation cost *TTC* and the carbon emissions of vehicles during transportation *TFU* in the transportation stage are calculated as follows:

$$TTC = \sum_{f=1}^{F}\left(C_3 \times K_f + \sum_{i=1}^{n_f}\sum_{j=1}^{n_f}\sum_{k=1}^{K_f} d_{ij}x_{ijk}\right) \tag{13}$$

$$FU_{ij} = hMV\lambda d_{ij} / v_{ij} + \gamma\lambda\alpha d_{ij}(W + w_i) + \beta\gamma\lambda d_{ij}v_{ij}^2. \tag{14}$$

$$TFU = \sum_{f=1}^{F}\sum_{i=1}^{n_f}\sum_{j=1}^{n_f}\sum_{k=1}^{K_f} x_{ijk}FU_{ij} \tag{15}$$

In Equation (13), C_3 represents the fixed dispatch cost of the vehicle. The value adopted in this paper is 180 yuan/vehicle. In Equation (14), FU_{ij} represents the carbon emissions of a vehicle when traveling from point i to point j, and its parameter values are referenced from literature [9]. In the IDFSP_VRPSPDFD problem addressed in this study, the optimization objectives are to minimize the total cost (TC) and minimize the total carbon emissions (TE), calculated as follows:

$$Min\ \ TC = TPC + TTC \tag{16}$$

$$Min\ \ TE = TFU + TEC \tag{17}$$

3 HH_ACO for IDFSP_MCVRPSDDFP

HH_ACO is a hybrid algorithm that combines ACO and HHA. In this study, the HHA algorithm is employed to explore the solution space of the IDFSP_MCVRPSDDFP problem, while the ACO algorithm is utilized to enhance the performance of the HHA algorithm. In HHA_ACO, the combination sequence of six neighborhood operations is optimized using the ACO algorithm to obtain a high-level population. The individuals in the high-level population serve as independent heuristic algorithms to control the low-level individuals for local search, thereby achieving exploration of the solution space.The structure diagram of HH_ACO is shown in **Fig. 1**.

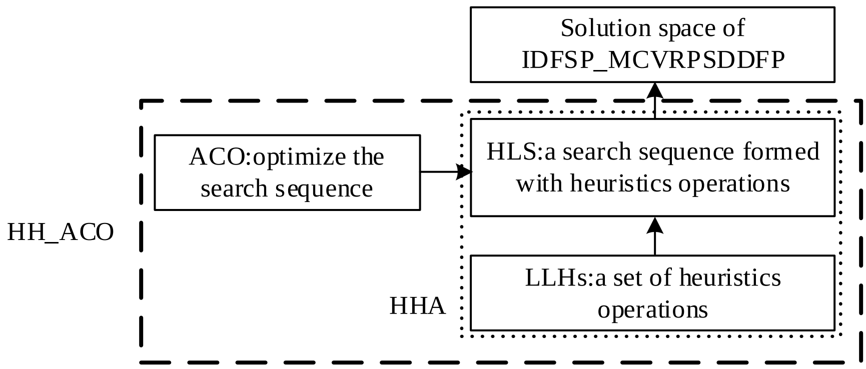

Fig. 1. Structure diagram of HH_ACO

3.1 Encoding and Decoding

In HH_ACO, each individual in the lower-level population represents a feasible solution to the original problem. How the solution is represented is a critical aspect of the optimization algorithm.

In this study, a random encoding approach is utilized to generate the total job sequence π, which is subsequently evenly allocated among all factories.Using a twodimensional array, the components required for processing at each factory are stored. This array consists of F rows, with each row containing a portion of the job sequence. The sequence of jobs represents the processing order at the factory. Simultaneously calculating the cost and carbon emissions of the products in the manufacturing and transportation phases provides the fitness of the low-level individual.

For the high-level population, each individual is composed of six different LLHs. By sequentially applying the LLHs of the high-level population to the individuals of the low-level population, the change in individual objective values represents the fitness of that high-level individual.

3.2 LLHs

In order to search for higher-quality solutions in the solution space of the IDFSP_VRPSPDFD problem, this study designed six LLHs based on the sequence $\Pi\left(\pi,\pi_f,\pi_f^k\right)$. The designed LLHs are as follows:

LLH1: Randomly select two points from sequence Π and swap the data of these two points.

LLH2: Randomly select two points from sequence Π and insert the point from the rear position before the point in the front position.

LLH3: Randomly select two points from sequence Π and reverse the data between these two points.

LLH4: Randomly select one point from sequence Π and place it at the first or last position of the sequence.

LLH5: Randomly select three points from sequence Π and move the data of the last two points to the position before the first point.

LLH6: Randomly select two points from sequence Π and move the data of these two points to the first two positions of the sequence.

Taking sequence $\Pi=[6,3,2,1,4,5]$ as an example, the six designed operations are shown in the following **Fig. 2**.

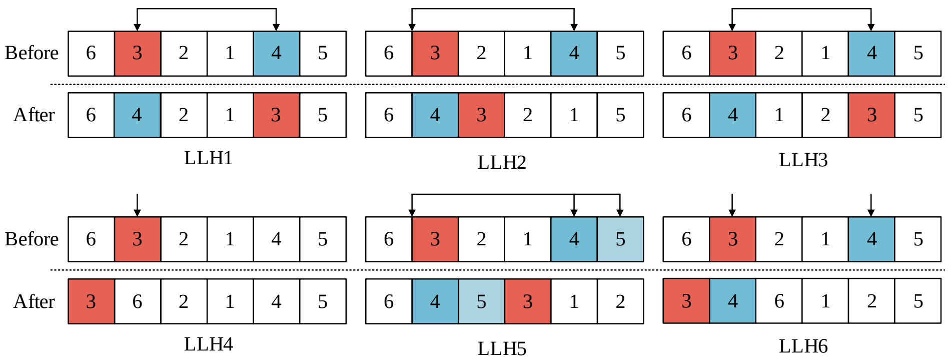

Fig. 2. LLHs on $\Pi=[6,3,2,1,4,5]$

3.3 Algorithm Procedure

Based on the above description, the flow of the HH_ACO algorithm is shown in **Fig. 3**. In this paper, the population size is set to $popsize$. The length of each high-level individual consists of n low-level heuristic operations. If $\max gen$ is set as the termination condition, then the time complexity of HH_ACO is $O(\max gen \times n \times popsize)$.

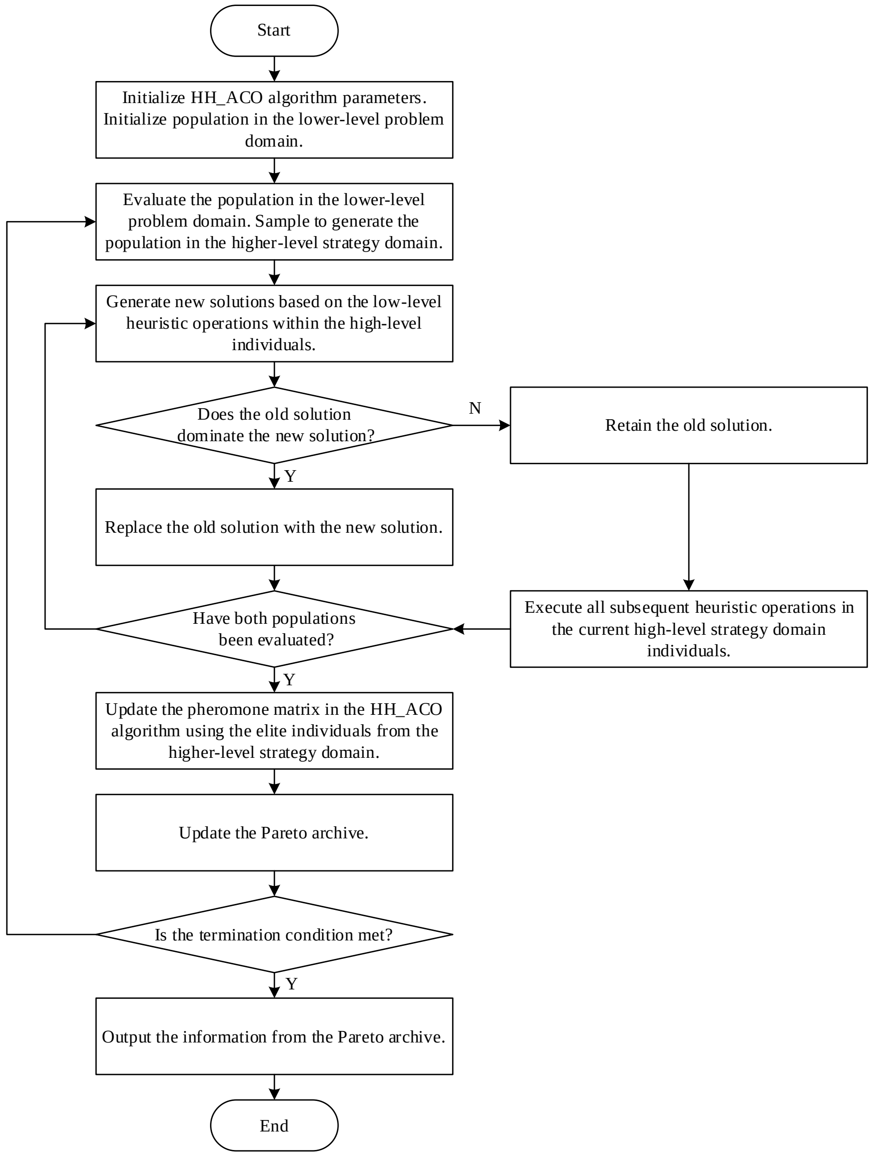

Fig. 3. HH_ACO flow chart

4 Experiment Analysis

All algorithms designed in this paper were implemented or reproduced using MATLAB R2022b. The computational experiments are executed on a PC with Inter(R) Core (TM) i5-12400 CPU @ 2.5 GHz processor and 16G of RAM under Microsoft Windows 10 OS. At the same time, employ the approach from references [10-12] to generate the dataset for this study. The parameter settings for HH_AC0 are as follows: population size $popsize = 40$; pheromone heuristic factor $\alpha = 0.1$; pheromone evaporation coefficient $\rho = 0.9$; pheromone intensity $Q = 1$.

To validate the effectiveness and superiority of the proposed HH_ACO, we selected well-established and widely used multi-objective optimization algorithms, namely MOEA/D [14], NSGAII [13] and SPEAII [15] for comparison. To ensure fairness in the experiments, all algorithms were configured with the termination criterion set as $N \times M \times F \times 0.1s$ and each instance was run 20 times independently.

Simultaneously, Inverted Generational Distance (IGD)[14] and Hypervolume (HV) [17] are used to evaluate the effectiveness of the proposed algorithm. IGD and HV are metrics used to assess the quality of the non-dominated solution set obtained by algorithms, accounting for both the convergence and the distribution of the achieved non-dominated solutions. A smaller IGD and a larger HV indicate better distribution and diversity of the results obtained by the algorithm. The calculation methods for IGD and HV are shown in Equations (18) and (19), respectively:

$$IGD = \frac{\sum_{i=1}^{num\_PF} |d_i|}{num\_PF} \tag{18}$$

$$HV(P,Z) = volume\left(\bigcup_{p \in P}^{|P|} (p, Z) \right) \tag{19}$$

In the equation (18), d represents the distance from the points on the reference solution set to the solutions obtained by each algorithm, num_PF represents the number of points in the reference solution set. Equation (19) represents the sum of the areas formed between the solutions obtained by the algorithm and the reference point. Here, P denotes the solutions obtained by each algorithm, and Z is the reference point, which is set to (1.0, 1.0) in this paper. The obtained results are shown in **Table 2**. For clarity, the best value in each row is highlighted in bold.

As shown in **Table 2**, it is clear that the proposed HH_ACO algorithm can achieve better results on datasets of different scales. This is because conventional multi-objective intelligent algorithms use a fixed sequence of local search operations during iterative loops to explore the solution space, without considering the impact of the operation sequence on search capability. This limitation affects the algorithm's ability to search the solution space effectively.

In HH_ACO, the ACO component adopts a high-level strategy to learn the heuristic operation sequences from high-quality individuals and dynamically mixes various low-level heuristic operators to update the population. This approach allows for effective exploration of the solution space, enabling the discovery of high-quality solutions in different regions of the solution space and enhancing the algorithm's search capability. Consequently, the HH_ACO algorithm performs exceptionally well across test cases of varying scales.

Table 2. Comparison of IGD and HV of HH_ACO, NSGA-II, MOEAD, and SPEAII.

F, M, N	SPEAII		NSGA-II		MOEAD		HH_ACO	
	IGD	HV	IGD	HV	IGD	HV	IGD	HV
2,10,100	**0.106**	**0.804**	0.278	0.442	0.413	0.27	0.106	0.764
3,20,100	0.224	0.577	0.225	0.539	0.396	0.268	**0.102**	**0.83**
4,10,100	0.13	0.602	0.209	0.502	0.343	0.279	**0.103**	**0.767**
5,10,100	0.112	**0.814**	0.302	0.493	0.439	0.236	**0.089**	0.781
2,5,20	0.177	0.628	0.265	0.488	0.379	0.283	**0.072**	**0.861**
3,5,20	0.167	0.678	0.22	0.574	0.381	0.298	**0.138**	**0.802**
2,5,50	0.142	0.705	0.301	0.44	0.424	0.235	**0.069**	**0.857**
3,5,50	0.183	0.661	0.237	0.589	0.406	0.308	**0.122**	**0.856**
4,5,50	0.208	0.646	0.237	0.549	0.454	0.243	**0.107**	**0.869**
6,10,72	**0.091**	**0.841**	0.347	0.461	0.524	0.241	0.101	0.798
5,10,75	0.186	0.608	0.211	0.559	0.315	0.366	**0.095**	**0.853**
2,10,80	0.192	0.59	0.266	0.475	0.333	0.338	**0.092**	**0.843**
3,10,80	**0.094**	**0.828**	0.265	0.538	0.351	0.372	0.094	0.81

5 Conclusions

This paper proposes a hyper-heuristic ant colony optimization (HH_ACO) algorithm to solve the integrated distributed permutation flow-shop problem and multiple compartments vehicle routing problem with simultaneous deterministic delivery and fuzzy pickup (IDFSP_MCVRPSDDFP), with the objectives of minimizing total cost and total carbon emissions. Firstly,the mixed integer linear programming model of IDFSP_MCVRPSPDFD is proposed. Secondly, the composition of the HH_ACO algorithm is introduced, which applies the ACO algorithm in the high-level strategy domain population of the HHA algorithm, leveraging the positive feedback principle of ACO. Additionally, six LLHs are designed to achieve in-depth search of the problem solution space. Finally, the effectiveness of the proposed HH_ACO is validated through examples of different scales. For future research, enhancing the operational speed of HH_ACO could be explored through reinforcement learning or machine learning techniques to obtain better solutions in a shorter time.

Acknowledgements. This research was supported by the National Natural Science Foundation of China (62173169 and 61963022) and the Basic Research Key Project of Yunnan Province (202201AS070030).

References

1. Moons S, Ramaekers K, Caris A ,et al. Integrating production scheduling and vehicle routing decisions at the operational decision level: A review and discussion. Computers & Industrial Engineering, 104, 224-245 (2017).
2. Karaoglan I, Kesen S E . The coordinated production and transportation scheduling problem with a time-sensitive product: a branch-and-cut algorithm. International Journal of Production Research, 55(2), 1-22 (2017).
3. Guo Z, Zhang D, Leung S Y S, et al. A bi-level evolutionary optimization approach for integrated production and transportation scheduling. Applied Soft Computing, 42, 215-228 (2016).
4. Zou, Xuxia, et al. "A coordinated algorithm for integrated production scheduling and vehicle routing problem. "International Journal of Production Research 56(15), 5005-5024 (2018).
5. Yang, Shengluo, and Zhigang Xu. "The distributed assembly permutation flowshop scheduling problem with flexible assembly and batch delivery." International Journal of Production Research 59(13), 4053-4071 (2021).
6. Naderi B, Ruiz R. The distributed permutation flowshop scheduling problem. Computers & operations research, 37(4), 754-768 (2010).
7. Colorni, Alberto, et al. "Heuristics from nature for hard combinatorial optimization problems." International Transactions in Operational Research 3(1), 1-21 (1996).
8. Shi, Yong, Toufik Boudouh, and Olivier Grunder. "A hybrid genetic algorithm for a home health care routing problem with time window and fuzzy demand." Expert Systems with Applications, 72, 160-176 (2017).
9. Guo N, Qian B, Na J, et al. A three-dimensional ant colony optimization algorithm for multi-compartment vehicle routing problem considering carbon emissions. Applied Soft Computing, 127, 109326 (2022).
10. Naderi, Bahman, and Rubén Ruiz. "The distributed permutation flowshop scheduling problem."Computers & operations research 37(4), 754-768 (2010).
11. Multiple Depot VRP Instances Vehicle Routing Problem, http://neo.lcc.uma.es/vrp/vrp-instances/multiple-depot-vrp-instances/, Last Accessed 2022/06/13.
12. Dethloff, Jan. "Vehicle routing and reverse logistics: The vehicle routing problem with simultaneous delivery and pick-up*:*OR-spektrum 23, 79-96 (2001).
13. Deb, Kalyanmoy, et al. "A fast and elitist multiobjective genetic algorithm: NSGA-II." IEEE transactions on evolutionary computation 6(2), 182-197 (2002).
14. Zhang, Qingfu , and H. Li . "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition." IEEE Transactions on Evolutionary Computation 11(6), 712-731 (2008).
15. Zitzler E, Laumanns M, Thiele L. SPEA2: Improving the strength pareto evolutionary algorithm. Technical Report Gloriastrasse, (2008).
16. Coello, Carlos A. Coello, and Nareli Cruz Cortés. "Solving multiobjective optimization problems using an artificial immune system." Genetic programming and evolvable machines 6, 163-190 (2005).
17. Zitzler E , Thiele L .Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach [J]. IEEE Transactions on Evolutionary Computation, 3(4):257-271 (1999).