IJSOM International Journal of Supply and Operations Management

Optimal Solving Multi-Depot Vehicle Routing Problems with Pickup-Delivery Requests and Time Window Constraints using Cooperative HHO-ATS Algorithm

Supaporn Suwannarongsri [1]✉

1. Corresponding author, Department of Materials Handling and Logistics Engineering, Faculty of Engineering, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand. E-mail: supaporn.s@eng.kmutnb.ac.th

Article Info

Article type:
Research Article

Article history:
Received June 14, 2026
Received in revised form July 2, 2026
Accepted August 21, 2026
Available online August 29, 2026

***Keywords*:**
multi-depot vehicle routing problems
pickup-delivery requests
time window constraints
cooperative HHO-ATS algorithm
metaheuristic optimization techniques

ABSTRACT

Objective: The multi-depot vehicle routing problem with pickup-delivery requests and time window constraints (MDVRPPDTW) is a complex NP-hard optimization problem in logistics and material handling. This study aims to develop an effective optimization approach for obtaining high-quality solutions to large-scale MDVRPPDTW instances.

Methods: Cooperative Harris Hawks Optimization-Adaptive Tabu Search (Cooperative HHO-ATS) algorithm is proposed. The algorithm employs a serial-heterogeneous framework that combines the global exploration capability of Harris Hawks Optimization (HHO) with the local exploitation strength of Adaptive Tabu Search (ATS). HHO is first used to identify promising regions of the search space, after which ATS refines the elite solution generated by HHO to improve solution quality and convergence. The proposed method was evaluated on ten benchmark MDVRPPDTW instances containing 200-400 customer locations.

Results: The performance of Cooperative HHO-ATS was compared with Genetic Algorithm (GA), Particle Swarm Optimization (PSO), standalone HHO, and ATS. Experimental results show that the proposed algorithm consistently achieves shorter total travel distances across all benchmark instances.

Conclusion: The integration of HHO and ATS in Cooperative HHO-ATS effectively balances exploration and exploitation, resulting in superior optimization performance. Cooperative HHO-ATS algorithm outperforms GA, PSO, HHO, and ATS in solving large-scale MDVRPPDTW instances and represents a promising approach for practical logistics routing optimization problems.

Cite this article: Suwannarongsri, S. (2026). Optimal Solving Multi-Depot Vehicle Routing Problems with Pickup-Delivery Requests and Time Window Constraints using Cooperative HHO-ATS Algorithm. *International Journal of Supply and Operations Management*, *XX*(X), pages. https://doi.org/10.22034/ijsom.2026.137299.3585

© Author(s) retain the copyright.
Publisher: Kharazmi University
DOI: https://doi.org/10.22034/ijsom.2026.137299.3585

1. Introduction

In 1959, the vehicle routing problem (VRP) was firstly introduced as one of the classical and fundamental problems in material handling and logistics engineering (Dantzig & Ramser, 1959). For mathematically speaking, the VRP is formulated as a combinatorial optimization problem and classified as the NP-hard problem due to its inherent computational complexity (Dantzig & Ramser, 1959; Golden & Assad, 1988; Bektas, 2006). The classical VRP has served as the foundation for the development of numerous extended variants that more accurately reflect real-world logistics operations including the multi-vehicle routing problem (MVRP), the capacitated VRP (CVRP), the VRP with time windows (VRPTW), the heterogeneous fleet VRP (HVRP), the VRP with pickup and delivery (VRPPD), and the multiple depots VRP (MDVRP) (Cordeau et al., 2002; Ramos et al., 2020). Among these variants, the multi-depot vehicle routing problem with pickup-delivery requests and time windows (MDVRPPDTW) represents the generalized formulation of the VRP that most closely resembles practical scenarios (Nagy & Salhi, 2005; Sombunthama & Kachitvichyanukul, 2010; Bai et al., 2026). This is because the MDVRPPDTW simultaneously addresses multiple depots, pickup and delivery operations, and predefined delivery time windows, requiring vehicles to both collect goods from customers and deliver goods to customers within specified time constraints.

Solving the VRP and its variants using exact or deterministic methods may fail to obtain the best solution or may require prohibitively long computational times, particularly for large-scale problem instances. This limitation is because of the characteristic of the NP-hard problems (Dantzig & Ramser, 1959; Golden & Assad, 1988; Bektas, 2006; Cordeau et al., 2002; Ramos et al., 2020). Consequently, many researchers have adopted approximation methods based on the metaheuristic optimization techniques to address the VRP and its variants. Although approximation methods do not guarantee global optimality, they are capable of producing high-quality solutions that are close to the near-global solutions within a reasonable computational time. Following the literature, metaheuristic optimization techniques used for solving the MVRP and MDVRP problems included neural network (NN) (Modares et al., 1999), variable neighborhood search (VNS) (Polacek et al., 2004), genetic algorithm (GA) (Zhou & Li, 2010), particle swarm optimization (PSO) (Sombuntham & Kachitvichayanukul, 2010), tabu search (TS) (Escobar et al., 2014), ant colony optimization (ACO) (Toklu et al., 2014), iterated local search (ILS) (Juan et al., 2014), cuckoo search (CS) (Goli et al., 2018), artificial bee colony (ABC) (George & Binu, 2018), evolutionary game analysis (EGA) (Yang et al., 2020), flower pollination algorithm (FPA) (Suwannarongsri, 2020), sector combination optimization (SCO) (Shi et al., 2020), parallel cuckoo search (PCS) (Suwannarongsri, 2021), large neighborhood search (LNS) (Schaap ett al., 2022), modern metaheuristic algorithm (MoMA) (Suwannarongsri, 2023), rat swarm optimization (RSO) based on LNS (Wei et al., 2024) and parallel grey wolf optimizer (PGWO) (Suwannarongsri, 2025). Metaheuristics conducted for optimizing the MDVRPPDTW problems included TS (Li & Lim, 2003), PSO (Ai & Kachitvichyanukul, 2009), GA (Alaia et al., 2015), adaptive LNS (ALNS) (Naccache et al., 2018), simulated annealing (SA) (Ghobadi et al., 2021), VNS (Ghobadi et al., 2021), ACO (Lv et al., 2025), hybrid bat-ALNS (Anass et al., 2025) and membrane-inspired evolutionary algorithm (Bai et al., 2026).

Despite the significant progress achieved in applying metaheuristic optimization techniques to the MDVRPPDTW, several research gaps remain. Most existing studies have focused on developing individual metaheuristic algorithms or conventional hybrid approaches, each exhibiting distinct advantages and limitations in balancing global exploration and local exploitation. Population-based algorithms generally possess strong exploration capabilities but may suffer from premature convergence or insufficient local refinement during the later stages of the search process. In contrast, trajectory-based algorithms provide efficient local exploitation but are often sensitive to the quality of the initial solution and may become trapped in local optima. These limitations become more pronounced when solving large-scale MDVRPPDTW instances involving hundreds of customer requests, where the search space increases exponentially. Moreover, only a limited number of studies have investigated cooperative optimization frameworks that systematically integrate complementary population-based and trajectory-based metaheuristics to effectively exploit the strengths of both search mechanisms for solving the MDVRPPDTW.

Motivated by these research gaps, this paper proposes a Cooperative Harris Hawks Optimization-Adaptive Tabu Search (Cooperative HHO-ATS) algorithm for solving the MDVRPPDTW. The proposed algorithm introduces a serial heterogeneous cooperative search framework that combines the global exploration capability of Harris hawks optimization (HHO) (Heidari et al., 2019) with the intensive local exploitation capability of adaptive tabu search (ATS) (Sujitjorn et al., 2006). Unlike conventional hybrid algorithms that simply embed one search strategy into another, the proposed Cooperative HHO-ATS enables HHO and ATS to cooperate sequentially, allowing promising solutions generated during the exploration phase to be systematically refined during the exploitation phase. This cooperative search strategy is expected to improve the balance between exploration and exploitation, accelerate convergence toward high-quality solutions, reduce the risk of premature convergence, and enhance solution quality for large-scale MDVRPPDTW instances. This means that the contribution of this paper lies in integrating these two algorithms (HHO and ATS) to enhance the optimization performance for solving MDVRPPDTW problems, rather than proposing a new metaheuristic optimization algorithm.

Based on the proposed optimization framework, the Cooperative HHO-ATS algorithm is applied to solve ten benchmark MDVRPPDTW instances consisting of 200-400 customer locations selected from the literature. The performance of the proposed algorithm is comprehensively evaluated by comparing with those obtained from GA, ATS, PSO, and HHO algorithms in terms of solution quality and computational performance. After an introduction proposed in Section 1, the remainder of this paper is organized as follows. The problem formulation including the MDVRPPDTW model, the objective and constrained functions are illustrated in Section 2. Cooperative HHO-ATS algorithm for the MDVRPPDTW problem optimization is described in Section 3. Experimental results are provided in Section 4. Discussion is presented in Section 5. Finally, conclusion and future research are given in Section 6.

2. Problem Formulation

The mathematical formulation of the MDVRPPDTW problem, including the objective function and the associated set of constraints, is described in this section. As discussed earlier, this problem belongs to the class of NP-hard combinatorial optimization problems in logistics and transportation. It represents an extension of the classical VRP by incorporating multiple depots, a fleet of vehicles, paired pickup-delivery requests, and strict service time window requirements.

In the MDVRPPDTW problem considered in this paper, several depots located at different geographical points dispatch a set of vehicles to serve customer requests. Each request involves collecting a particular item from a designated pickup location and transporting it to a corresponding delivery location using the same vehicle. The pickup activity must always occur before the delivery operation for the same request. Vehicles start their routes from assigned depots, visit the required service locations, and return to their originating depots once all assigned tasks have been completed. Furthermore, each customer node must be served exactly once by one vehicle in the fleet. Both pickup and delivery services must be performed within specified time windows. In addition, several operational restrictions are imposed, including vehicle capacity limits, availability of vehicles in the fleet, and maximum allowable duration of each route.

The optimization objective of the MDVRPPDTW is to minimize the total travel distance of all vehicles. Achieving this objective helps reduce overall transportation and operational costs while ensuring that all customer demands are fulfilled and all constraints of the problem are satisfied. As an illustrative example, consider an instance of the MDVRPPDTW that contains 2 depots, 40 customer or service nodes (excluding the depots), and 6 available vehicles. This example is illustrated in Figure 1, where ○ represents customer nodes (service locations), ■ indicates the depots D1 and D2, [,] stands for the time window associated with each service, and — denotes the routes traveled by vehicles for performing pickup and delivery operations.

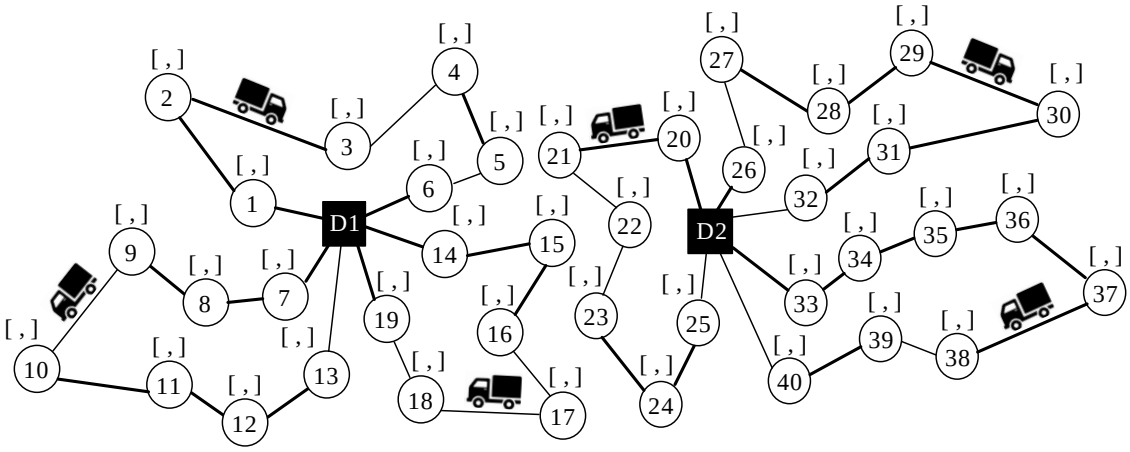


Figure 1. MDVRPPDTW problem consisting of 2 depots, 40 service locations, and 6 vehicles.

Prior to establishing the mathematical formulation of the MDVRPPDTW, some assumptions are defined to clarify the problem setting.

- Every vehicle route originates from a depot and must terminate at the same depot.
- The vehicle fleet is considered homogeneous, and all vehicles are initially stationed at the depots.
- Each customer location must be visited exactly once and serviced by a single vehicle.
- A vehicle cannot operate overlapping routes; therefore, its routes must be performed sequentially in time.
- The cumulative demand served along a vehicle's route must not exceed its capacity.
- The number of vehicles utilized in the solution cannot be greater than the maximum number of vehicles available in the fleet.
- The overall duration of any route, including both travel time and service time, must remain within a predefined upper bound.

Following the MDVRPPDTW model (Alaia et al., 2013; Alaia et al., 2015), the problem is represented by using a connected-directed graph $G = (N, E)$, where N denotes the set of vertices corresponding to depots and customer locations, while $A = \{(i,j \mid i,j \in N)\}$ represents the set of arcs that define the distances between nodes. The parameters and decision variables of the proposed MDVRPPDTW model are defined as described in Table 1.

The objective function $f_{\text{MDVRPPDTW}}$ is expressed in Equation (1), while the associated constraints are described in Equations (2)-(17).

Table 1. Set, parameters and decision variables of MDVRPPDTW model.

Natation	Definition
Sets	
D	Set of depots, $\{1,\ldots,Dep\}$
N	Set of nodes (pickup and delivery), $\{1,\ldots,n\}$
N_p	Set of pickup nodes, $\{1,\ldots,(n/2)\}$
N_d	Set of delivery nodes, $\{1,\ldots,(n/2)\}$
N_c	Set of couples: pickup and delivery, $\{1,\ldots,(n/2)\}$
V_m	Set of available vehicles from depot m, $\{V_1,\ldots,V_{Dep}\}$

Parameters	
K	The total number of vehicles available for all the depots
k	The number of vehicles, $\{1,\ldots,\lvert V_m\rvert\}$
Q	The maximum capacity of a vehicle
TE_i	The earliest time that node i can be serviced by a vehicle
TL_i	The latest permitted time to leave node i
TS_i	The service time at node i
TA_i	The arrival time of the assigned vehicle at node i
TD_i	The departure time of the vehicle from node i
TW_i	The waiting time of the vehicle at node i
$Dist_{max}$	The maximum distance that can be covered by vehicle tours
$[TE_i, TL_i]$	The time window of node i
C_i	Couple (f_p, f_d): pickup node (f_p) with its corresponding delivery node (f_c), $\forall i \in \{1,\ldots,(n/2)\}$
q_i	The request of the node i, (if $q_i < 0$, it is a delivery node; otherwise, it is a pickup node)
d_{ij}	The Euclidean distance between node i and j; ($d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$)
L_i^k	The lode of the vehicle k before leaving the node i
t_{ij}^k	The time taken by the vehicle k to travel from node i to node j
Valiables	
x_{ij}^{mk}	$x_{ij}^{mk} = \begin{cases} 1, & \text{if the vehicle } k \text{ is assigned to depot } m, \text{ and travels from node } i \text{ to node } j \\ 0, & \text{otherwise.} \end{cases}$, where $i,j \in N$, $m \in D$, and $k \in V_m$

Mathematical Formulation:

Minimize
$$f_{\text{MDVRPPDTW}} = \sum_{m\in D}\sum_{k\in V_m}\sum_{i\in N\cup m}\sum_{j\in N\cup m} d_{ij}x_{ij}^{mk} \tag{1}$$

Subject to
$$\sum_{i\in D\cup N}\sum_{j\in D\cup N} d_{ij}x_{ij}^{mk} \le Dist_{max}, \qquad \forall m \in D, k \in V_m \tag{2}$$

$$\sum_{m\in D}\sum_{i\in D\cup N}\sum_{k\in V_m} x_{ij}^{mk} = 1, \qquad \forall j \in D\cup N \tag{3}$$

$$\sum_{m\in D}\sum_{j\in D\cup N}\sum_{k\in V_m} x_{ij}^{mk} = 1, \qquad \forall i \in D\cup N \tag{4}$$

$$\sum_{j\in N} x_{ij}^{mk} = \sum_{j\in N} x_{ji}^{mk}, \qquad \forall i = m \in D, k \in V_m \tag{5}$$

$$x_{ij}^{mk} = 1 \Rightarrow L_i^k = 0, \qquad \forall i \in D, j \in N, k \in V_m \tag{6}$$

$$x_{ji}^{mk} = 1 \Rightarrow L_i^k = 0, \qquad \forall i \in D, j \in N, k \in V_m \tag{7}$$

$$x_{ij}^{mk} = 1 \Rightarrow L_j^k = L_i^k + q_j, \qquad \forall i, j \in N, k \in V_m \tag{8}$$

$$0 < L_i^k \le Q, \qquad \forall i \in N, k \in V_m \tag{9}$$

$$x_{ij}^{mk} = 1 \Rightarrow TA_j = TD_i + t_{ij}^k, \qquad \forall k \in V_m \tag{10}$$

$$TE_i > TA_i \Rightarrow TW_i = TE_i - TA_i, \qquad \forall i \in N \tag{11}$$

$$TD_i = TA_i + TS_i, \qquad \forall i \in N \tag{12}$$

$$TD_i = TS_i = 0, \qquad \forall i \in D \tag{13}$$

$$TE_i \le TA_i \le TL_i, \qquad \forall i \in N \tag{14}$$

$$TE_i \le TD_i \le TL_i, \qquad \forall i \in N \tag{15}$$

$$TA_i \le TL_i + TS_i, \qquad \forall i \in N \tag{16}$$

$$TD_{f_p} < TD_{f_d}, \qquad \forall i \in N_c, f_p \in N_p, f_d \in N_d \tag{17}$$

The objective function $f_{\text{MDVRPPDTW}}$ as stated in Equation (1) aims to minimize the total travel distance, while satisfying the constraints presented in Equations (2)-(17). Constraint in Equation (2) restricts the total distance of each vehicle route so that it does not exceed the predefined maximum limit. Constraints in Equations (3) and (4) ensure that every service node is visited exactly once and is assigned to only one vehicle. Constraint in Equation (5) requires each vehicle to start and finish its route at the same depot. The vehicle capacity conditions are represented by constraints in Equations (6) and (7), which ensure that vehicles depart from and return to the depot without any load. Constraint in Equation (8) specifies that vehicle k traveling from node i to node j carries a load when leaving the depot. Constraint in Equation (9) ensures that the load transported by vehicle k never exceeds its maximum vehicle capacity limits. Time-related conditions are described in constraints in Equations (10)-(16). For each node i, the time window $[TE_i, TL_i]$ defines the permissible period during which service must take place. Constraint in Equation (10) determines the arrival time at node j, and service at each customer must begin within the specified time window. If a vehicle arrives before the earliest allowable time, it must wait until service can begin. Constraint in Equation (11) represents the waiting time, whereas constraint in Equation (12) specifies the departure time from node i. Constraint in Equation (13) determines the service and departure times at each depot. Constraints in Equations (14) and (15) ensure that the arrival and departure times at node i occur within the allowable time window, meaning they cannot be earlier than the earliest time or later than the latest departure time. Constraint in Equation (16) guarantees that the arrival time at node i does not exceed the sum of the latest departure time and the service duration. Finally, constraint in Equation (17) represents the precedence constraints between pickup and delivery tasks, which requires that the pickup node f_p of each customer pair C_i must be visited before its corresponding delivery node f_c. From the MDVRPPDTW model expressed in Equations (1)-(17), it can be easily extended or adapted to represent other variants of the VRP including the CVRP, VRPTW, HVRP, VRPPD and MDVRP problems.

3. Cooperative HHO-ATS Algorithm for MDVRPPDTW Optimization

In this section, the ATS and HHO algorithms are briefly introduced. Then, Cooperative HHO-ATS algorithm is proposed. Subsequently, the optimization procedure for the MDVRPPDTW based on Cooperative HHO-ATS is described in detail as follows.

3.1 ATS Algorithm

The original ATS was firstly proposed in 2006 (Sujitjorn et al., 2006). It consists of three main mechanisms: the tabu list (TL), adaptive radius (AR), and backtracking (BT). The ATS employs a stochastic neighborhood search framework as a trajectory-based metaheuristic. In each iteration, a set of feasible candidate solutions is randomly generated within a specified search radius to form the neighborhood. The TL acts as a short-term memory that records previously

explored solutions to avoid revisiting the same regions and to enhance search diversity. The AR mechanism dynamically adjusts the search radius to strengthen local exploration. As the radius decreases, the search becomes more focused on promising areas of the solution space, which improves convergence speed and solution quality. When the algorithm is trapped in a local optimum, the BT mechanism is activated. It randomly selects previously visited solutions stored in the TL to redirect the search toward unexplored regions (Sujitjorn et al., 2006). The pseudo code of the ATS is presented in Figure 2.

ATS algorithm

- Initialize the objective function $f(\boldsymbol{x})$, $\boldsymbol{x} = (x_1,\ldots,x_d)^T$, and search spaces, search radius R,
  weighting factor α, $(0<\alpha<1)$, and TL, iteration t =1 and maximum iteration t_{max} as TC.
- Randomly generate the initial best solution $\boldsymbol{x}^*$ and evaluate $f(\boldsymbol{x}^*)$.
while ($t \le t_{max}$ or TC are not met)
- Randomly N neighborhood solutions around $\boldsymbol{x}^*$ within R.
- Evaluate all N solutions via $f(\boldsymbol{x}^*)$.
- Let one of the best solutions among N be $\boldsymbol{x}$.
    if1 ($f(\boldsymbol{x})<f(\boldsymbol{x}^*)$)
        Update $\boldsymbol{x}^* = \boldsymbol{x}$.
    end if1
        Store $\boldsymbol{x}$ in TL and activate AR mechanism using $R = \alpha R$.
    if2 (Local entrapment occur)
        - Invoke BT mechanism by randomly selecting $\boldsymbol{x}'$ (one of the solutions stored in TL),
        - Update $\boldsymbol{x}^* = \boldsymbol{x}'$.
    end if2
- Update $t = t+1$.
end while
Report the best solution $\boldsymbol{x}^*$.

Figure 2. Pseudo code of ATS algorithm.

From Figure 2, $f(\cdot)$ stands for the objective function, while $\boldsymbol{x} = (x_1,\ldots,x_d)$ represents the candidate solution in a d-dimensional search space, d denotes the problem dimension, R stands for the search radius, α denotes the weighting factor associated with the AR mechanism, t is the iteration counter, t_{max} specifies the maximum number of iterations, serving as the termination criterion (TC). Furthermore, $\boldsymbol{x}^*$ represents the current best solution identified during the search process. The set N denotes the neighbourhoods comprising feasible candidate solutions generated at each iteration, and $\boldsymbol{x} \in N$ corresponds to the best solution within N. The notation $\boldsymbol{x}'$ indicates one of the previously visited solutions stored in the TL and potentially utilized during the BT mechanism.

The computational complexity of the ATS, expressed by Big-O notation, is $O(NTD+1) \approx O(NTD)$, where N is the number of neighborhood solutions generated in each iteration, T denotes the maximum number of iterations, and D represents the problem dimension. For further details about ATS, readers can refer to references (Sujitjorn et al., 2006; Puangdownreong et al., 2004; Kulworawanichpong et al., 2004).

3.2 HHO Algorithm

In 2019, the original HHO was firstly proposed (Heidari et al., 2019). It was derived from the hunting patterns of Harris hawks as a population-based metaheuristic. Hawks utilize collaborative and adaptive tactics to identify, chase, and capture prey under varying environmental conditions. Accordingly, the HHO algorithm simulates this behavior through three sequential stages: exploration, transition between exploration and exploitation, and exploitation (Heidari et al., 2019).

In the exploration phase, the hawk positions are updated according to Equation (18). Let $X(t)$ be the positions of a hawk at iteration t, and $X(t+1)$ be its updated positions. The parameters r_1, r_2, r_3, r_4, and q are uniformly distributed random numbers in [0, 1]. LB and UB denote the lower and upper bounds of the search space, respectively. $X_{rand}(t)$ represents a randomly selected hawk, while $X_{prey}(t)$ denotes the prey position, corresponding to the best solution found so far. The population mean position $X_m(t)$ is calculated by Equation (19), where $X_i(t)$ is the position of the i-th hawk and N_h is the population size.

$$X(t+1)=\begin{cases}X_{rand}(t)-r_1\,|\,X_{rand}(t)-2r_2X(t)\,|, & q\geq 0.5\\ [X_{prey}(t)-X_m(t)]-r_3[LB+r_4(UB-LB)], & q<0.5\end{cases} \tag{18}$$

$$X_m(t)=\frac{1}{N_h}\sum_{i=1}^{N_h}X_i(t) \tag{19}$$

The transition from exploration to exploitation is governed by the prey's escaping energy E, as defined in Equation (20). Here, T denotes the maximum number of iterations and E_0 is the initial escaping energy, randomly generated within [-1, 1] to represent stochastic prey behaviour. As the iterations increase, the magnitude of E gradually decreases, enabling a dynamic search mechanism. When $|E|\geq 1$, the algorithm performs global exploration, whereas $|E|<1$ shifts the search toward exploitation.

$$E=2E_0\left(1-\frac{t}{T}\right) \tag{20}$$

In the exploitation phase, the hawks' chasing behaviour and the prey's escape actions are modelled through four strategies: soft besiege, hard besiege, soft besiege with progressive rapid dives, and hard besiege with progressive rapid dives. The soft besiege occurs when $|E|\geq 0.5$ and $r\geq 0.5$, where hawks gradually encircle the prey and update their positions using Equations (21)-(22), with $J = 2(1-r_5)$ representing the prey's random jump. The hard besiege is triggered when $|E|<0.5$ and $r\geq 0.5$, indicating that the prey has lost most of its escaping energy. Then, the hawk positions are updated using Equation (23).

$$X(t+1)=\Delta X(t)-E\,|\,JX_{prey}(t)-X(t)\,| \tag{21}$$

$$\Delta X(t)=X_{prey}(t)-X(t) \tag{22}$$

$$X(t+1)=X_{prey}(t)-E\,|\,\Delta X(t)\,| \tag{23}$$

When $r<0.5$, rapid dive strategies are applied to simulate surprise pounces. If $|E|\geq 0.5$, the soft besiege with progressive rapid dives is executed, where hawks first form a loose encirclement and then perform Lévy-flight-based dives to refine the search. The hawk positions are updated via the objective function $F(\cdot)$ according to Equation (24), while the candidate moves Y and Z are generated using Equations (25)-(26). From Equation (26), S is a random vector by size $1\times D$, D is the problem dimension, and LF denotes the Lévy-flight function calculated by Equation (27), where u and v stand for random values within [0, 1], β denotes a default constant value, σ stands for the standard deviation, and Γ is the standard Gamma function. When $|E|<0.5$, the hard besiege with progressive rapid dives is activated, where Y is computed using Equation (28) and the mean position $X_m(t)$ is defined by Equation (19). The pseudo code of HHO is illustrated in Figure 3.

$$X(t+1)=\begin{cases}Y, & F(Y)<F(X(t))\\ Z, & F(Z)<F(X(t))\end{cases} \tag{24}$$

$$Y = X_{prey}(t) - E \mid JX_{prey}(t) - X(t) \mid \tag{25}$$

$$Z = Y + S \times LF(D) \tag{26}$$

$$LF(X) = 0.01\frac{u \times \sigma}{|v|^{1/\beta}}, \quad \sigma = \left[\frac{\Gamma(1+\beta) \times \sin(\pi\beta/2)}{\Gamma[(1+\beta)/2] \times \beta \times 2^{(\beta-1)/2}}\right]^{1/\beta} \tag{27}$$

HHO algorithm

- Initialize the objective function $F(\boldsymbol{x})$, $\boldsymbol{x} = (x_1,\ldots,x_d)^T$, search spaces, population size of hawks N_h, iteration $t = 1$, and maximum iterations T as TC.
- Randomly generate population X_i ($i = 1,2,\ldots,N_h$).
while ($t \le T$ or TC are not met)
- Calculate $F(X_i)$ of all N_h hawks.
- Set X_{prey} as the current best location.
for (each hawk (X_i))
    - Update E_0 and jump strength J using E_0 = 2rand()-1, J = 2[1-rand()].
    - Update the energy E using Equation (20).
    if1 ($|E| \ge 1$), % *Exploration phase*
        - Update positions of hawks using Equation (18).
    end if1
    if2 ($|E| < 1$), % *Exploitation phase*
        if ($r \ge 0.5$ and $|E| \ge 0.5$), % *Soft besiege*
            - Update positions of hawks using Equation (21).
        else if ($r \ge 0.5$ and $|E| < 0.5$), % *Hard besiege*
            - Update positions of hawks using Equation (23).
        else if ($r < 0.5$ and $|E| \ge 0.5$), % *Soft besiege with progressive rapid dives*
            - Update positions of hawks using (24), when Y using Equation (25) and Z using Equation (26).
        else if ($r < 0.5$ and $|E| < 0.5$), % *Hard besiege with progressive rapid dives*
            - Update positions of hawks using Equation (24), when Y using Equation (28) and Z using Equation (26).
    end if2
end for
- Update X_{prey} and $t = t + 1$.
end while
Report the best solution X_{prey}.

Figure 3. Pseudo code of HHO algorithm.

$$Y = X_{prey}(t) - E \mid JX_{prey}(t) - X_m(t) \mid \tag{28}$$

The computational complexity of the HHO, expressed using Big-O notation, is $O(N\times(T+TD+1)) \approx O(N\times(T+TD))$, where N denotes the number of hawks, T is the maximum number of iterations, and D represents the problem dimension, (Heidari et al., 2019; Huang et al., 2023). For more details about HHO, readers can refer to references (Heidari et al., 2019; Alabool et al., 2021; Tripathy et al., 2022; Hussien et al., 2022; Huang et al., 2023).

3.3 Cooperative HHO-ATS Algorithm

Cooperative HHO-ATS algorithm is developed as a serial-heterogeneous metaheuristic algorithm. It combines HHO and ATS algorithms for balancing global exploration and local exploitation. Cooperative HHO-ATS operates through two sequentially coordinated phases. In the first phase, the HHO algorithm as illustrated in Figure 3 is executed to perform global exploration across the entire search domain, leveraging its strong diversification capability to identify

high-quality elite feasible solutions. In the second phase, the ATS algorithm as illustrated in Figure 2 is subsequently activated, utilizing the elite solution obtained from HHO as the initial starting point. This phase emphasizes intensive local exploitation to accelerate convergence toward the global optimum.

The transition between these two phases is controlled by a predefined switching iteration (SI), whose value depends on the characteristics of the optimization problem. For the normal case, SI may be set to approximately 50% of the maximum number of iterations. The overall procedure of Cooperative HHO-ATS algorithm can be represented by the flow diagram shown in Figure 4.

As mentioned earlier, the computational complexity of ATS is $O(NTD)$, whereas that of HHO is $O(N\times(T+TD))$. Based on the definition of Big-O notation, $O(N(T+TD))$ can be simplified to $O(NT(D+1))$, which is asymptotically equivalent to $O(NTD)$. Therefore, the computational complexity of the proposed Cooperative HHO-ATS algorithm is $O(ND(T_{HHO}+T_{ATS}))$, where N is the number of hawks (or candidate solutions), D denotes the problem dimension, T_{HHO} is the maximum number of iterations of the HHO phase, and T_{ATS} is the maximum number of iterations of the ATS phase. Since the proposed algorithm consists of two sequential optimization phases, the overall computational complexity is the sum of the complexities of HHO and ATS, i.e., $O(NT_{HHO}D)+ O(NT_{ATS}D) = O(ND(T_{HHO}+T_{ATS}))$.

Although the proposed Cooperative HHO-ATS algorithm consists of two sequential optimization phases, its computational complexity increases linearly with respect to the total number of iterations ($T_{HHO}+T_{ATS}$). Therefore, the proposed cooperative framework preserves polynomial-time complexity and remains computationally scalable for large-scale optimization problems.

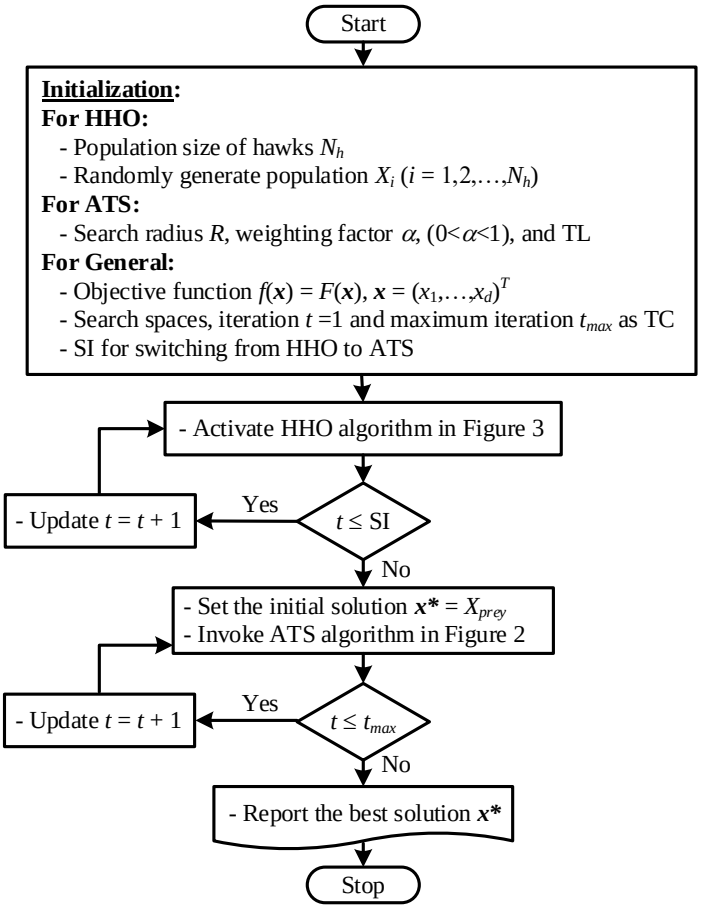


Figure 4. Flow diagram of cooperative HHO-ATS algorithm.

3.4 MDVRPPDTW Optimization using Cooperative HHO-ATS Algorithm

The MDVRPPDTW optimization based on Cooperative HHO-ATS can be described step-by-step as follows.

Step-0 *For MDVRPPDTW*: initialize the objective function $f_{\text{MDVRPPDTW}}(\cdot)$ of the MDVRPPDTW as stated in Equation (1) and the constrained functions as expressed in Equations (2)-(17), as well as their corresponding parameters (D, N_p, N_d, N_c, C_i, V_m, Q, t_{ij}^k, L_i^k, TE_i, TL_i, TS_i, TA_i, TD_i, TW_i, and $Dist_{max}$), also initialize the search spaces Θ of control variables (N, d_{ij} and K) as the problem solutions.
For HHO: initialize the population size of hawks N_h, and randomly generate population X_i (i = 1,2,…,N_h).
For ATS: initialize the search radius R, weighting factor α, (0<α<1), and TL.
For General: initialize the initial iteration t =1, the maximum number of iterations t_{max} as the termination criteria (TC), and SI for switching from HHO to ATS.

Stop-1 If $t \le$ SI, go to Step-2 to activate HHO. Otherwise, go to Step-7 to invoke ATS.

Stop-2 Evaluate all hawks X_i via the objective function $f_{\text{MDVRPPDTW}}(\cdot)$ in Equation (1) and the constrained functions in Equations (2)-(17). Select X_i giving minimum $f_{\text{MDVRPPDTW}}(\cdot)$ as X_{prey})current best location(.

Stop-3 For each hawk X_i, update the initial energy E_0 = 2rand-()1 and jump strength J = 2]1-rand[(). Then, update the escaping energy of the prey E using Equation) 20(.

Stop-4 If $|E| \ge 1$ (*Exploration phase*), update the location vector using Equation) 18(and go to Step-6. Otherwise, go to Step-5.

Stop-5 If $|E| < 1$ (*Exploitation phase*),

Step-5.1 If $r \ge 0.5$ and $|E| \ge 0.5$ (*Soft besiege*), update X_i by using Equation) 21(.
Otherwise, go to Step-5.2.

Step-5.2 If $r \ge 0.5$ and $|E| < 0.5$ (*Hard besiege*), update X_i by using Equation) 23(.
Otherwise, go to Step-5.3.

Step-5.3 If $r < 0.5$ and $|E| \ge 0.5$ (*Soft besiege with progressive rapid dives*), update X_i by using Equation)24(, when Y is obtained by Equation) 25(and Z is calculated by Equation) 26(.
Otherwise, go to Step-5.4.

Step-5.4 If $r < 0.5$ and $|E| < 0.5$ (*Hard besiege with progressive rapid dives*), update X_i by using Equation)24(, once Y is calculated by Equation) 28(and Z is obtained by Equation) 26(.

Stop-6 Evaluate all X_i via the objective function $f_{\text{MDVRPPDTW}}(\cdot)$ in Equation (1) and the constrained functions in Equations (2)-(17). Update X_{prey} and $t = t$+1. Then, go back to Step-1 for executing a next iteration.

Stop-7 Update $\boldsymbol{x}^* = X_{prey}$ as the initial solution of ATS.

Stop-8 Randomly N neighborhood solutions around $\boldsymbol{x}^*$ within R.

Stop-9 Evaluate all N solutions via the objective function $f_{\text{MDVRPPDTW}}(\cdot)$ in Equation (1) and the constrained functions in Equations (2)-(17). Let one of the best solutions among N giving minimum $f_{\text{MDVRPPDTW}}(\cdot)$ be $\boldsymbol{x}$.

Stop-10 If $f(\boldsymbol{x}) < f(\boldsymbol{x}^*)$, update $\boldsymbol{x}^* = \boldsymbol{x}$. Otherwise, go to Step-11.

Stop-11 Store $\boldsymbol{x}$ in TL, and activate AR mechanism by using $R = \alpha R$.

Stop-12 If local entrapment occurs, invoke BT mechanism by randomly selecting $\boldsymbol{x}'$ (one of the solutions stored in TL). Update $\boldsymbol{x}^* = \boldsymbol{x}'$. Otherwise, go to Step-13.

Stop-13 If $t \le t_{max}$, update $t = t$+1. Then, go back to Step-8 for proceeding a next iteration. Otherwise, terminate the search process and report the best solution found $\boldsymbol{x}^*$.

4. Experimental Results

According to the literature, the real-world MDVRPPDTW problems were collected by Li and Lim (Li & Lim, 2001; Li & Lim, 2003; Li & Lim, 2008) for instance categories with approximately 100, 200, 400, 600, 800, and 1000 locations. In this work, ten benchmark MDVRPPDTW instances consisting of approximately 200-400 locations are selected from Li and Lim's collection to test the performance of Cooperative HHO-ATS algorithm, as summarized in Table 2.

Table 2. Ten selected MDVRPPDTW instances consisting of 200-400 locations.

MDVRPPDTW instances	Original names (Li & Lim)	Number of locations (N)	Number of depots (D)	Number of vehicles (m)	Vehicle capacity (Q)
MDVRPPDTW#1	lc1-2-1	212	5	50	200
MDVRPPDTW#2	lc1-2-3	206	4	50	200
MDVRPPDTW#3	lc2-2-1	204	4	50	700
MDVRPPDTW#4	lrc2-2-1	202	4	50	1,000
MDVRPPDTW#5	lrc2-2-3	202	4	50	1,000
MDVRPPDTW#6	lc1-4-1	422	7	100	200
MDVRPPDTW#7	lc1-4-3	420	7	100	200
MDVRPPDTW#8	lc2-4-1	406	6	100	700
MDVRPPDTW#9	lr2-4-1	402	6	100	1,000
MDVRPPDTW#10	lr2-4-3	404	6	100	1,000

In order to optimize the MDVRPPDTW problems by using Cooperative HHO-ATS algorithm, a symmetric distance assumption, $d_{ij} = d_{ji}$, is adopted. Depot locations are arbitrarily defined. Each vehicle is constrained to a maximum daily working time of 8 hours, with an average travel speed of approximately 80 km/h. Consequently, the total travel distance for each vehicle must not exceed 640 km per day. The dataset of ten selected MDVRPPDTW instances, including pickup-delivery requests and time window constraints as well as the precedence constraints between pickup and delivery tasks, is adopted from Li and Lim's benchmark (Li & Lim, 2008). In this study, the performance of Cooperative HHO-ATS algorithm for solving the MDVRPPDTW problems is compared with those of GA, ATS, PSO, and HHO. All algorithms are implemented in MATLAB (R2018b) and executed on a computer equipped with an Intel(R) Core(TM) i5-3470 CPU @ 3.60 GHz and 4.0 GB of RAM. For each MDVRPPDTW instance, 100 independent trial-runs are performed to obtain the best solution.

The search parameters of Cooperative HHO-ATS algorithm are set based on the preliminary study on the MDVRPPDTW instances by varying N_h = 5, 10, 15,…,60, N = 5, 10, 15,…,60, R = 5%, 10%, 25%,…,50% of search bounds, $J_{\max}$ = 5, 10, 15,…,40 (for BT machanism), and SI = 200, 250, 300,…,800. The best parameters for most MDVRPPDTW instances are N_h = 30-50, N = 25-50, R = 15-25% of search bounds, $J_{\max}$ = 10-20, and SI = 350-600. Therefore, in all MDVRPPDTW instances, the search parameters will be fixed, i.e., N_h = 40, N = 40, R = 20% of search bounds, $J_{\max}$ = 15 (for BT machanism), and SI = 450. In addition, t_{max} = 2,000 is set as the TC for all instances. Following the preliminary study, the parameters in the AR mechanism are adjusted in three stages, i.e., Stage-I: if Δf < 1e–2, $R = R\times$1e–3, Stage-II: if Δf < 1e–3, $R = R\times$1e–4 and Stage-III: if Δf < 1e–4, $R = R\times$1e–5, where $\Delta f = |f(\mathbf{x}^*) - f(\mathbf{x}')|$. For comparison with GA, ATS, PSO, and HHO, the search parameters of these algorithms are set to ensure a fair comparison, as follows.

For GA (Holland, 1975; Goldberg, 1989), number of populations is set to 40 (fixed as N and N_h of Cooperative HHO-ATS), the crossover probability is 0.95, and the mutation probability is 0.05. The maximum number of generations, *Max_Gen*, is set to 2,000 (fixed as t_{max} of Cooperative HHO-ATS).

For ATS (Sujitjorn et al., 2006), number of neighborhood members is N = 40 (fixed as N of Cooperative HHO-ATS), and $J_{\max}$ = 15. In the AR mechanism, three stages are applied, i.e., Stage-I: if Δf < 1e–2, $R = R\times$1e–3, Stage-II: if Δf < 1e–3, $R = R\times$1e–4 and Stage-III: if Δf < 1e–4, $R = R\times$1e–5. The maximum number of iterations, *Max_Iter*, is set to 2,000 (fixed as t_{max} of Cooperative HHO-ATS).

For PSO (Kennedy & Eberhart, 1995; Eberhart & Shi, 2001), number of swarms is set to 40 (fixed as N and N_h of Cooperative HHO-ATS), the cognitive learning rate is 2.0, and the social learning rate is 2.0. The maximum number of generations, *Max_Gen*, is set to 2,000 (fixed as t_{max} of Cooperative HHO-ATS).

For HHO (Heidari et al., 2019), number of hawks is N_h = 40 (fixed as N_h of Cooperative HHO-ATS) and t_{max} = 2,000 (fixed as t_{max} of Cooperative HHO-ATS).

As results obtained over 100 trial-runs, the best convergence curves of GA, ATS, PSO, HHO, and Cooperative HHO-ATS for the MDVRPPDTW#1 instance are depicted in Figure 5 as an example. The convergence curves of other MDVRPPDTW instances are omitted because they exhibit patterns similar to those shown in Figure 5. According to Figure 5, it can be observed that Cooperative HHO-ATS demonstrates superior performance, achieving better solutions and a faster convergence rate than GA, ATS, PSO, and HHO. Based on the overall experimental results, the optimal solutions for ten selected MDVRPPDTW instances obtained by GA, ATS, PSO, HHO, and Cooperative HHO-ATS are summarized in Table 3. For example, MDVRPPDTW#1 has 212 service locations and 5 depots. Its depot and service locations are plotted in Figure 6(a), whereas its optimal tours obtained by GA, ATS, PSO, HHO, and Cooperative HHO-ATS are illustrated in Figures 6(b)-6(f), respectively.

To investigate the contribution of each component of the proposed Cooperative HHO-ATS algorithm based on the results obtained from 100 independent trial-runs, an ablation study is conducted by comparing three variants: HHO-only, ATS-only, and the complete Cooperative HHO-ATS algorithm, as summarized in Table 3. The HHO-only variant is used to evaluate the contribution of the population-based exploration mechanism, whereas the ATS-only variant is used to assess the contribution of the trajectory-based exploitation mechanism. The comparison results demonstrate the individual effects of HHO and ATS and highlight the effectiveness of their cooperation within the proposed framework. The results indicate that neither HHO nor ATS alone can achieve the performance of the complete Cooperative HHO-ATS algorithm. HHO provides effective global exploration capability; however, its obtained solutions may require further refinement through local exploitation. Conversely, ATS improves solution refinement but is highly dependent on the quality of the initial solution. By sequentially integrating HHO and ATS, the proposed algorithm effectively combines exploration and exploitation mechanisms, resulting in improved solution quality and convergence performance.

Referring to results obtained over 100 trial-runs, the statistical details of the results obtained by the GA, ATS, PSO, HHO, and Cooperative HHO-ATS algorithms, including the best, worst, mean, standard deviation (SD), and median values in kilometers (km), as well as the average CPU time (Ave-CPUT) in seconds (s), are reported in Tables 4-8. These statistical results are also presented as boxplots in Figure 7.

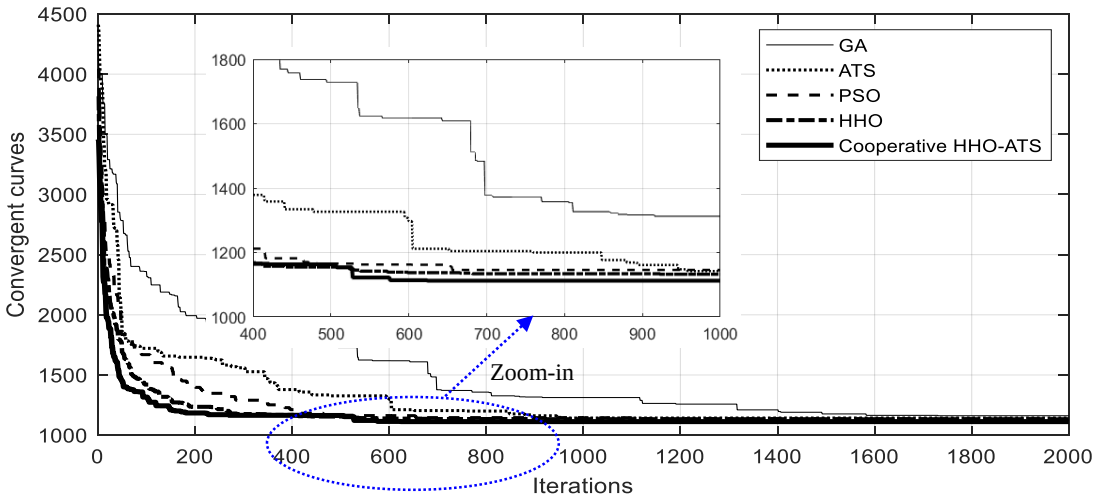


Figure 5. Convergent curves of all algorithms for MDVRPPDTW#1.

Table 3. Optimal solutions of MDVRPPDTW instances.

MDVRPPDTW instances	Optimal solutions* (Li & Lim)		No. of *D*	No. of *K*	Optimal tour (km.)				
	No. of *K*	Optimal tour (km.)			GA	ATS	PSO	HHO	Cooperative HHO-ATS
MDVRPPDTW#1	20	2,704.54	5	15	1,166.45	1,146.99	1,137.52	1,133.07	1,126.20
MDVRPPDTW#2	17	3,127.78	4	12	2,106.38	1,984.70	1,878.65	1,836.49	1,818.56
MDVRPPDTW#3	6	1,931.44	4	8	894.76	860.12	854.07	828.61	814.43
MDVRPPDTW#4	6	3,595.18	4	12	2,464.54	2,316.05	2,150.83	2,112.95	1,920.84
MDVRPPDTW#5	4	2,881.99	4	8	1,687.07	1,624.82	1,618.40	1,576.33	1,514.02
MDVRPPDTW#6	40	7,152.06	7	28	4,426.22	4,269.43	4,012.85	3,894.82	3,682.18
MDVRPPDTW#7	32	8,678.23	7	21	4,554.96	4,375.38	4,165.13	3,946.07	3,853.95
MDVRPPDTW#8	12	4,116.33	6	12	2,578.40	2,450.61	2,342.86	2,310.12	2,274.51
MDVRPPDTW#9	8	9,726.88	6	15	4,606.12	4,458.78	4,315.77	4,281.54	4,264.60
MDVRPPDTW#10	5	10,176.94	6	18	6,682.73	6,273.94	5,898.01	5,411.25	5,108.36

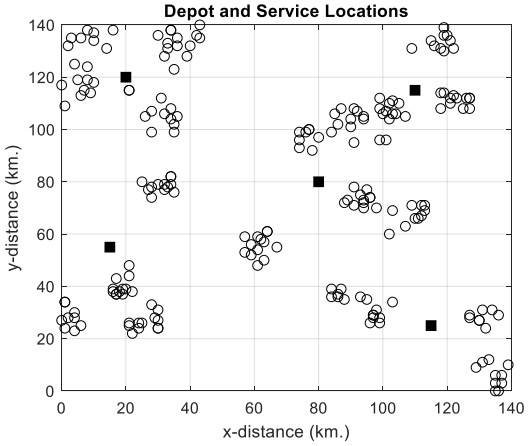
Note: * Optimal solutions of Li & Lim represented for MDVRPPDTWs having only one depot.

Table 4. Statistical results of MDVRPPDTW instances obtained by GA.

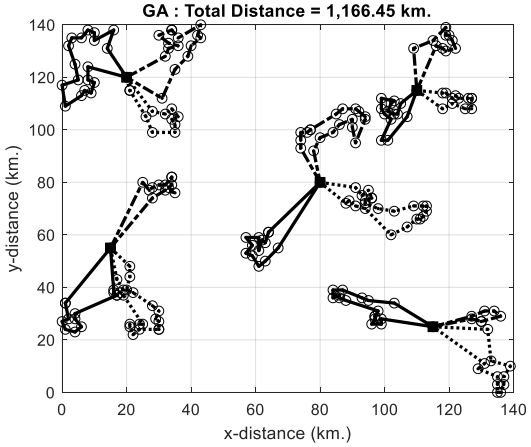
Statistical results	MDVRPPDTW#1	MDVRPPDTW#2	MDVRPPDTW#3	MDVRPPDTW#4	MDVRPPDTW#5
Best	1,166.45	2,106.38	894.76	2,464.54	1,687.07
Worst	1,792.41	3,486.03	1,398.74	3,834.02	2,688.95
Mean	1,486.02	2,828.72	1,158.42	3,159.68	2,184.03
SD	174.17	421.54	143.95	403.01	286.73
Median	1,478.23	2,890.69	1,173.03	3,182.18	2,196.84
Ave-CPUT	3,245.06	3,148.12	3,112.67	3,087.41	3,080.32
Statistical results	**MDVRPPDTW#6**	**MDVRPPDTW#7**	**MDVRPPDTW#8**	**MDVRPPDTW#9**	**MDVRPPDTW#10**
Best	4,426.22	4,554.96	2,578.40	4,606.12	6,682.73
Worst	7,899.31	8,380.68	4,074.22	8,476.08	10,973.24
Mean	6,256.76	6,565.57	3,270.51	6,652.87	8,850.16
SD	1,037.83	1,112.70	446.75	1,226.09	1,272.59
Median	6,207.27	6,550.46	3,307.58	6,822.13	8,880.37
Ave-CPUT	6,446.95	6,407.82	6,182.54	6,125.65	6,159.72

Table 5. Statistical results of MDVRPPDTW instances obtained by ATS.

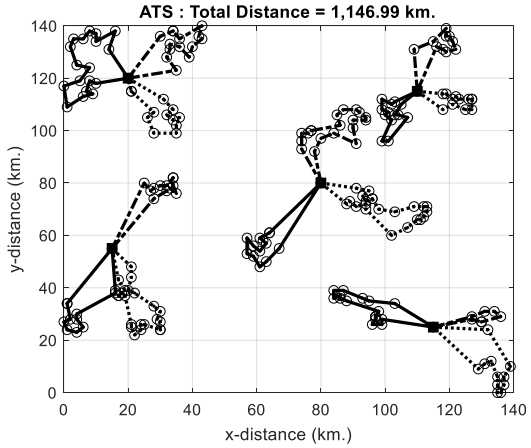
Statistical results	MDVRPPDTW#1	MDVRPPDTW#2	MDVRPPDTW#3	MDVRPPDTW#4	MDVRPPDTW#5
Best	1,146.99	1,984.70	860.12	2,316.05	1,624.82
Worst	1,547.86	3,099.41	1,194.77	3,494.58	2,482.20
Mean	1,365.12	2,491.83	1,039.86	2,940.14	2,059.11
SD	122.67	334.41	99.59	339.04	241.37
Median	1,373.54	2,487.67	1,048.64	2,947.87	2,058.43
Ave-CPUT	2,184.58	2,118.09	2,092.70	2,077.36	2,070.17
Statistical results	**MDVRPPDTW#6**	**MDVRPPDTW#7**	**MDVRPPDTW#8**	**MDVRPPDTW#9**	**MDVRPPDTW#10**
Best	4,269.43	4,375.38	2,450.61	4,458.78	6,273.94
Worst	7,192.77	7,563.12	3,876.83	7,877.40	9,994.82
Mean	5,823.88	5,995.11	3,180.67	6,080.34	8,112.43
SD	834.09	897.29	419.37	940.29	1,071.67
Median	5,875.31	5,947.88	3,130.02	6,039.89	8,119.77
Ave-CPUT	4,336.93	4,306.76	4,153.41	4,115.56	4,139.78



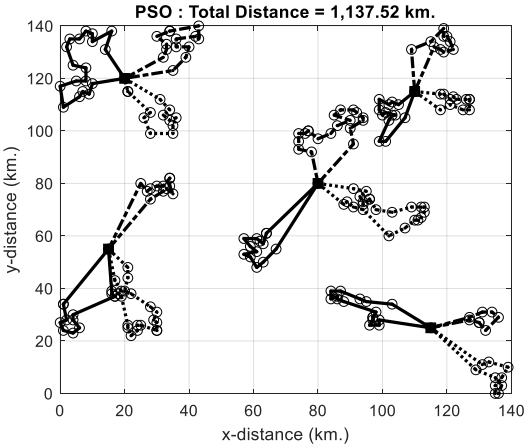
(a) Depot and service locations.



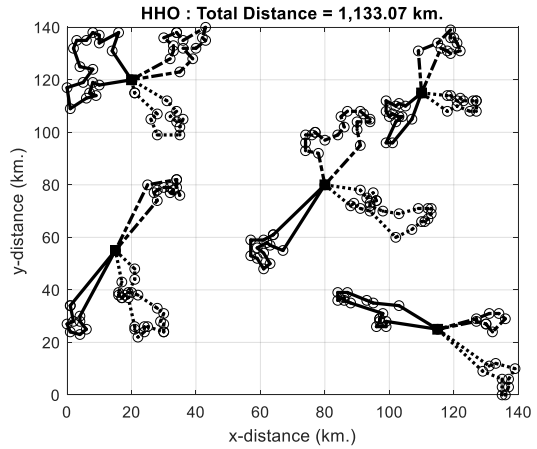
(b) Optimal tour by GA.



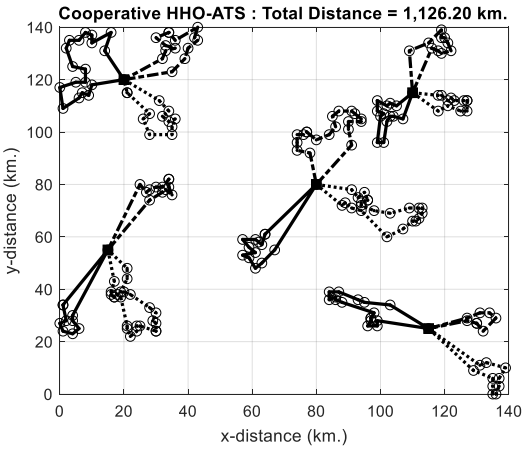
(c) Optimal tour by ATS.



(d) Optimal tour by PSO.



(e) Optimal tour by HHO.



(f) Optimal tour by Cooperative HHO-ATS.

Figure 6. Optimal tours of MDVRPPDTW#1.

Table 6. Statistical results of MDVRPPDTW instances obtained by PSO.

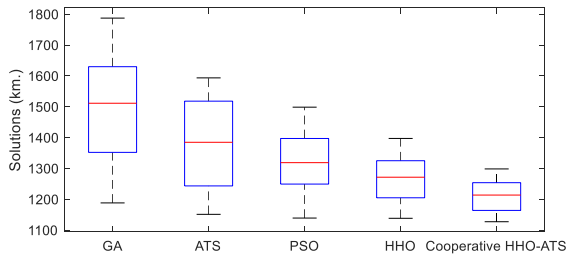
Statistical results	MDVRPPDTW#1	MDVRPPDTW#2	MDVRPPDTW#3	MDVRPPDTW#4	MDVRPPDTW#5
Best	1,137.52	1,878.65	854.07	2,150.83	1,618.40
Worst	1,458.84	2,675.92	1,099.23	2,893.79	2,384.76
Mean	1,288.30	2,232.58	987.10	2,546.80	2,016.51
SD	93.41	221.68	67.86	209.71	233.22
Median	1,280.93	2,207.67	987.66	2,558.98	2,011.52
Ave-CPUT	1,760.65	1,706.09	1,684.74	1,673.38	1,666.30
Statistical results	**MDVRPPDTW#6**	**MDVRPPDTW#7**	**MDVRPPDTW#8**	**MDVRPPDTW#9**	**MDVRPPDTW#10**
Best	4,012.85	4,165.13	2,342.86	4,315.77	5,898.01
Worst	6,682.14	6,899.44	3,555.82	7,284.46	8,993.52
Mean	5,280.83	5,589.87	2,975.27	5,818.02	7,555.04
SD	763.58	775.02	353.55	908.14	958.19
Median	5,227.20	5,643.86	2,983.04	5,740.01	7,610.28
Ave-CPUT	3,492.89	3,466.97	3,340.53	3,311.62	3,331.96

Table 7. Statistical results of MDVRPPDTW instances obtained by HHO.

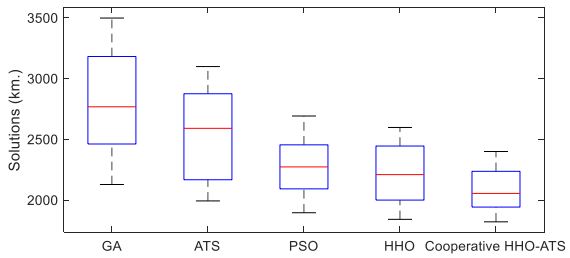
Statistical results	MDVRPPDTW#1	MDVRPPDTW#2	MDVRPPDTW#3	MDVRPPDTW#4	MDVRPPDTW#5
Best	1,133.07	1,836.49	828.61	2,112.95	1,576.33
Worst	1,396.81	2,590.77	1,047.12	2,799.45	2,293.58
Mean	1,267.68	2,227.48	940.47	2,486.90	1,961.04
SD	73.38	210.93	59.08	195.10	211.28
Median	1,264.36	2,226.72	943.07	2,506.38	1,963.47
Ave-CPUT	1,548.59	1,504.26	1,480.67	1,471.34	1,464.29
Statistical results	**MDVRPPDTW#6**	**MDVRPPDTW#7**	**MDVRPPDTW#8**	**MDVRPPDTW#9**	**MDVRPPDTW#10**
Best	3,894.82	3,946.07	2,310.12	4,281.54	5,411.25
Worst	5,885.76	6,492.68	3,411.75	7,244.49	7,964.34
Mean	4,909.03	5,235.48	2,849.33	5,736.27	6,747.71
SD	620.20	759.79	344.32	830.35	898.73
Median	4,925.02	5,280.61	2,845.42	5,809.04	6,805.68
Ave-CPUT	3,070.94	3,047.82	2,934.50	2,909.86	2,927.87

Table 8. Statistical results of MDVRPPDTW instances obtained by Cooperative HHO-ATS.

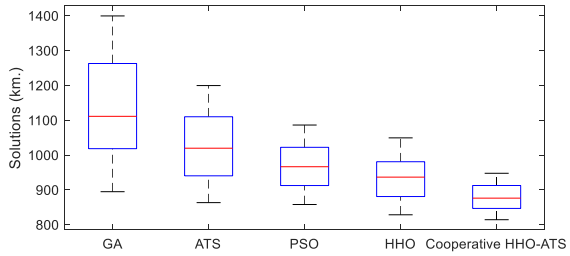
Statistical results	MDVRPPDTW#1	MDVRPPDTW#2	MDVRPPDTW#3	MDVRPPDTW#4	MDVRPPDTW#5
Best	1,126.20	1,818.56	814.43	1,920.84	1,514.02
Worst	1,298.87	2,398.26	944.79	2,486.12	2,091.38
Mean	1,213.26	2,117.81	873.32	2,206.47	1,801.64
SD	46.62	170.51	37.94	176.13	167.26
Median	1,213.92	2,129.58	868.41	2,226.90	1,801.83
Ave-CPUT	1,516.80	1,469.19	1,450.27	1,441.82	1,434.08
Statistical results	**MDVRPPDTW#6**	**MDVRPPDTW#7**	**MDVRPPDTW#8**	**MDVRPPDTW#9**	**MDVRPPDTW#10**
Best	3,682.18	3,853.95	2,274.51	4,264.60	5,108.36
Worst	5,377.41	5,896.82	3,245.03	6,696.22	7,799.12
Mean	4,581.29	4,896.87	2,761.22	5,397.58	6,487.29
SD	462.42	628.55	266.94	694.97	712.36
Median	4,555.64	4,858.73	2,744.18	5,362.71	6,517.27
Ave-CPUT	3,007.61	2,983.89	2,873.76	2,849.03	2,867.65



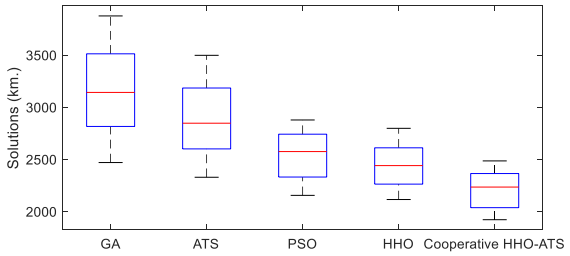
(a) MDVRPPDTW#1.



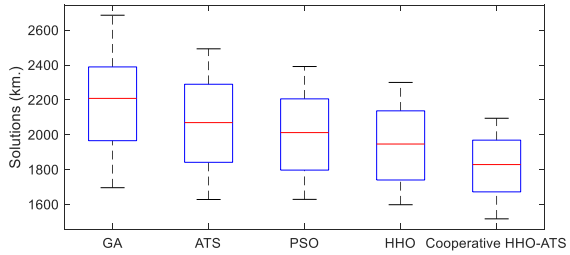
(b) MDVRPPDTW#2.



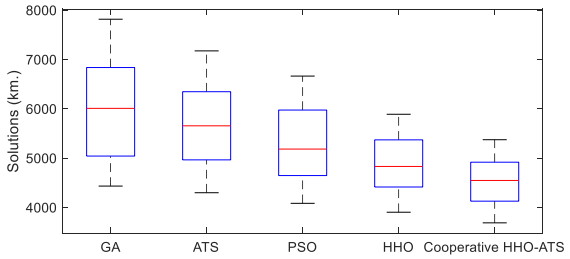
(c) MDVRPPDTW#3.



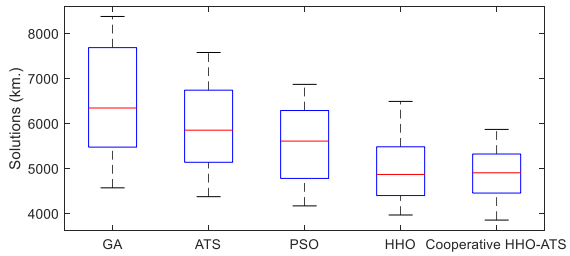
(d) MDVRPPDTW#4.



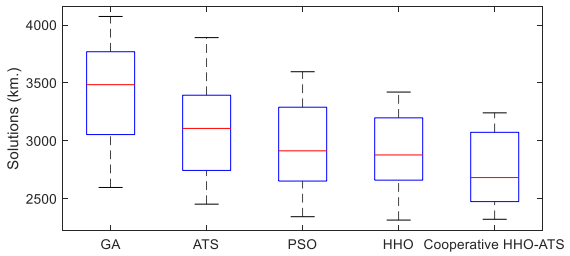
(e) MDVRPPDTW#5.



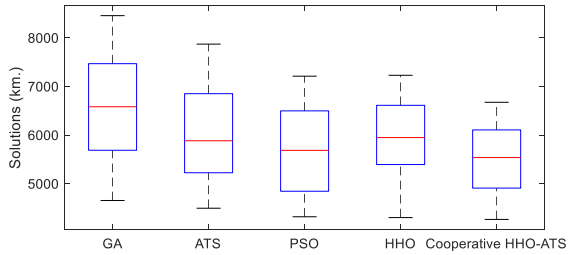
(f) MDVRPPDTW#6.



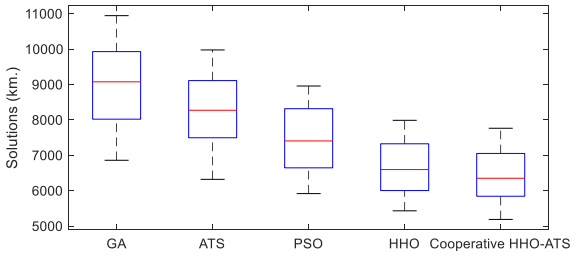
(g) MDVRPPDTW#7.



(h) MDVRPPDTW#8.



(i) MDVRPPDTW#9.



(j) MDVRPPDTW#10.

Figure 7. Boxplots of solutions obtained by GA, ATS, PSO, HHO, and Cooperative HHO-ATS algorithms.

To validate the algorithm's superiority and to investigate whether the performance differences among the compared algorithms are statistically significant, a two-sided Wilcoxon signed-rank test (Wilcoxon, 1945; Derrac et al., 2011) is conducted. The test is performed using the mean solutions obtained from the all MDVRPPDTW instances shown in Tables 4-8. Cooperative HHO-ATS is selected as the base model and compared separately with GA, ATS, PSO, and HHO algorithms. The following hypotheses are formulated:

H_0: There is no significant difference between the performance of Cooperative HHO-ATS and that of the compared algorithm.

H_1: There is a significant difference between the performance of Cooperative HHO-ATS and that of the compared algorithm.

The significance level is set at $\alpha = 0.05$. The results of the Wilcoxon signed-rank test are presented in Table 9.

Table 9. Results of Wilcoxon signed-rank test for all compared algorithms.

Algorithms	Average of mean results	Standard deviation (SD)	Base model for comparison	*Z*	*p* (>|*Z*|)
GA	4,241.27	2,623.77	Cooperative HHO-ATS	-2.8031	0.005062
ATS	3,908.85	2,403.82		-2.8031	0.005062
PSO	3,629.03	2,245.65		-2.8031	0.005062
HHO	3,436.14	2,040.61		-2.8031	0.005062
Cooperative HHO-ATS	3,233.68	1,945.89		-	-

5. Discussion

The experimental results illustrated in Figures 6(b)-6(f) and summarized in Table 3 demonstrate that Cooperative HHO-ATS consistently generated the shortest optimal tours across all MDVRPPDTW instances, followed by HHO, PSO, ATS, and GA, respectively. These findings suggest that the effectiveness of a metaheuristic algorithm for solving MDVRPPDTW depends not only on its exploration and exploitation capabilities but also on its ability to maintain an appropriate balance between these two search behaviors under a fixed TC.

Among the evaluated algorithms, GA exhibited the lowest solution quality. This result can be explained by the search mechanism of GA, which is based on evolutionary operators and genotype-to-phenotype transformations (Holland, 1975; Goldberg, 1989). As a population-based metaheuristic, GA relies heavily on uniformly distributed random processes to generate and evolve candidate solutions. Although these mechanisms support population diversity and global exploration, they may require a relatively large number of generations to effectively exploit promising regions of the search space. Consequently, when the search process is constrained by a limited number of generations (*Max_Gen*), GA may not sufficiently refine high-potential solutions, resulting in inferior solution quality when compared with the other algorithms considered in this study.

ATS achieved better solution quality than GA but remained inferior to PSO, HHO, and Cooperative HHO-ATS. As a trajectory-based metaheuristic, ATS is inherently exploitation-oriented and focuses on intensively exploring local neighborhoods of candidate solutions (Sujitjorn et al., 2006; Puangdownreong et al., 2014; Kulworawanichpong et al., 2014). While this characteristic enhances local search capability by using AR and BT mechanisms, the use of uniformly distributed randomization may limit its ability to diversify the search process and explore distant promising regions effectively. Under a fixed TC (*Max_Iter*), ATS tends to spend a substantial portion of the search effort within local search regions, increasing the likelihood of premature convergence and limiting its ability to achieve solution quality comparable to the higher-performing algorithms.

PSO outperformed both GA and ATS, indicating a stronger capability for exploring the global search space. As a population-based metaheuristic, PSO utilizes information sharing among particles to guide the search toward promising regions (Kennedy & Eberhart, 1995; Eberhart & Shi, 2001). This mechanism promotes exploration and helps avoid some of the local search limitations observed in ATS. However, the search behavior of PSO is still primarily driven by uniformly distributed stochastic processes, which may result in excessive emphasis on global

exploration while providing insufficient local refinement. Consequently, under a fixed TC (*Max_Gen*), PSO may not fully exploit high-quality regions of the search space, leading to solution quality that remains lower than that achieved by HHO and Cooperative HHO-ATS.

HHO produced better solutions than GA, ATS, and PSO, highlighting the effectiveness of its search strategy. Unlike the previous algorithms, HHO incorporates both exploration and exploitation mechanisms through adaptive prey-surrounding and attacking behaviors (Heidari et al., 2019; Alabool et al., 2021; Tripathy et al., 2022; Hussien et al., 2022). Furthermore, the integration of uniform randomization and Lévy-flight search enables HHO to diversify the search process while maintaining the ability to intensify the search around promising solutions. These characteristics improve search efficiency and allow HHO to identify high-quality solutions more effectively under a fixed TC (t_{max}). Nevertheless, despite its strong performance, the standalone HHO algorithm may still experience challenges in maintaining an optimal balance between global exploration and local exploitation throughout the entire search process. As a result, its solution quality remains slightly inferior to that of Cooperative HHO-ATS.

The superior performance of Cooperative HHO-ATS can be attributed to the complementary strengths of its constituent algorithms. Specifically, HHO contributes effective global exploration and adaptive search diversification, whereas ATS provides intensive local exploitation and solution refinement. The cooperative framework enables these search behaviors to interact synergistically rather than operate independently. Moreover, the optimal-predefined SI facilitates an adaptive transition between exploration and exploitation phases, allowing the search process to allocate computational effort more effectively according to the current search state. This balanced search behavior enhances search efficiency and improves the algorithm's ability to identify high-quality solutions within a fixed TC. Consequently, Cooperative HHO-ATS consistently achieved the shortest optimal tours across all MDVRPPDTW instances and outperformed the standalone GA, ATS, PSO, and HHO algorithms.

Besides solution quality, robustness is another important performance criterion for stochastic optimization algorithms. A robust algorithm is expected to produce consistent results over repeated runs, implying low sensitivity to randomness. Consequently, lower SD values correspond to higher robustness and lower sensitivity. Based on the SD values reported in Tables 4-8 and the boxplots in Figure 7, Cooperative HHO-ATS consistently obtains the smallest SD for all MDVRPPDTW instances. This demonstrates that Cooperative HHO-ATS is more robust and less sensitive to stochastic effects than GA, ATS, PSO, and HHO algorithms.

According to Table 9, the Wilcoxon signed-rank test revealed statistically significant differences between Cooperative HHO-ATS and all the compared algorithms. From Table 9, all comparisons produce *Z*-value of -2.8031 and *p*-value of 0.005062 in every pairwise comparison, which is lower than the significance level of 0.05. Therefore, the null hypothesis is rejected for all comparisons. These results indicate that the performance of Cooperative HHO-ATS differed significantly from those of GA, ATS, PSO, and HHO algorithms. Moreover, Cooperative HHO-ATS achieves a lower mean solution than every competing algorithm for all ten MDVRPPDTW instances. Because the MDVRPPDTW is a minimization problem, the results demonstrate that Cooperative HHO-ATS significantly outperformed all the compared algorithms.

Although the proposed Cooperative HHO-ATS algorithm demonstrates competitive performance for solving the MDVRPPDTW, several limitations should be acknowledged. First, the proposed algorithm consists of two sequential optimization phases, namely HHO-based exploration and ATS-based exploitation, which may increase computational time compared with single metaheuristic approaches. Second, the performance of the proposed algorithm still depends on several control parameters, including population size, iteration number, and neighborhood search settings. Appropriate parameter tuning is therefore required to achieve satisfactory performance for different problem characteristics. Third, the experimental evaluation is conducted using benchmark instances from the literature, which may not fully represent the complexity and uncertainty of real-world logistics operations. Factors such as dynamic customer requests, stochastic travel times, and changing transportation conditions are not considered in the current model. Finally, although the proposed algorithm can effectively solve large-scale MDVRPPDTW instances, further

investigation is required to evaluate its scalability for extremely large-scale problems and practical industrial applications.

6. Conclusion

Optimal solving the MDVRPPDTW problems using Cooperative HHO-ATS algorithm has been proposed in this paper. As a serial-heterogeneous optimization framework, the integration of HHO and ATS in Cooperative HHO-ATS algorithm was designed to balance global exploration and local exploitation. In the first stage, global exploration of the search space was performed using HHO. In the subsequent stage, ATS was employed, where the elite solution obtained from HHO was used as the initial solution to enhance local exploitation and accelerate convergence toward a near-optimal solution. Cooperative HHO-ATS has been evaluated against ten MDVRPPDTW instances with approximately 200-400 customer locations selected from the literature. A comparative analysis has been conducted and compared with GA, ATS, PSO, and HHO algorithms. The experimental results show that Cooperative HHO-ATS consistently achieves high-quality solutions across all instances, yielding shorter total distances than those obtained by other algorithms. Moreover, the Wilcoxon signed-rank test has been conducted to validate the algorithm's superiority. It can be concluded that Cooperative HHO-ATS outperforms GA, ATS, PSO, and HHO algorithms in MDVRPPDTW optimization.

For future development, several research directions can be considered to further enhance the capability and applicability of the proposed Cooperative HHO-ATS algorithm. First, adaptive parameter control and self-tuning mechanisms can be incorporated to reduce the dependency on manually selected algorithm parameters and improve the robustness of the search process for different problem characteristics. Second, advanced hybridization strategies, such as integrating machine learning-based prediction mechanisms or adaptive neighborhood structures, can be investigated to further improve the balance between exploration and exploitation. Third, the proposed algorithm can be extended to address more complex and dynamic MDVRPPDTW scenarios by considering practical factors, such as uncertain customer demands, time-dependent travel conditions, dynamic requests, and multi-objective optimization criteria. Finally, the applicability of Cooperative HHO-ATS can be further evaluated in a broader range of real-world industrial engineering problems, including production planning, logistics and supply chain management, job shop scheduling, assembly line balancing, and cutting stock problems.

Author Contributions

Supaporn Suwannarongsri is the sole author of this paper and is responsible for all aspects of the work, including Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing—original draft preparation, Writing—review and editing, and Visualization.

Data Availability Statement

Research data supporting this publication are available upon request.

Acknowledgements

The authors would like to thank King Mongkut's University of Technology North Bangkok, Bangkok, Thailand for financial support and anonymous referees for their constructive comments.

Funding

This paper was funded by King Mongkut's University of Technology North Bangkok with contract no. KMUTNB-69-KNOW-40.

Conflict of interest

The authors declare no conflict of interest.

References

Ai, T. J., & Kachitvichyanukul, V. (2009). Particle swarm optimization and two solution representations for solving the capacitated vehicle routing problem. *Computer & Industrial Engineering*, *56*(1), 380-387. https://doi.org/10.1016/j.cie.2008.06.012

Alabool, H. M., Alarabiat, D., Abualigah, L., & Heidari, A. A. (2021). Harris hawks optimization: a comprehensive review of recent variants and applications. *Neural Computing and Applications*, *33*, 8939-8980. https://doi.org/10.1007/s00521-021-05720-5

Alaia, E. B., Dridi, I. H., Bouchriha, H., & Borne, P. (2013). Optimization of the multi-depot & multi-vehicle pickup and delivery problem with time windows using genetic algorithm. In *International Conference on Control, Decision and Information Technologies (CoDIT)*. https://doi.org/10.1109/CoDIT.2013.6689568

Alaia, E. B., Dridi, I. H., Bouchriha, H., & Borne, P. (2015). Genetic algorithm for multi-criteria optimization of multi-depots pick-up and delivery problems with time windows and multi-vehicles. *Acta Polytechnica Hungarica*, *12*(8), 155-174. https://doi.org/10.12700/APH.12.8.2015.8.9

Anass, T., Said, E., Salim, E. B. M., Abdelouahed, A. I., & Najdi, L. (2025). Optimizing medical logistics networks: a hybrid bat-ALNS approach for multi-depot VRPTW and simultaneous pickup-delivery. *Journal of Electronics, Electromedical Engineering, and Medical Informatics, 7*(4), 991-1011. https://doi.org/10.35882/jeeemi.v7i4.1054

Bai, Z., Snášel, V., Mirjalili, S., Vo, B., Kong, L., & Wang, X. (2026). A novel membrane-inspired evolutionary algorithm framework for VRPTW. *Applied Intelligence, 57*. https://doi.org/10.1007/s10489-025-07068-y

Bektas, T. (2006). The multiple traveling salesman problem: an overview of formulations and solution procedures. *Omega*, *34*(3), 209-219. https://doi.org/10.1016/j.omega.2004.10.004

Cordeau, J-F., Gendreau, M., Laporte, G., Potvin, J-Y., & Semet, F. (2002). A guide to vehicle routing heuristics. *Journal of the Operational Research Society*, *53*(5), 512-522. https://doi.org/10.1057/palgrave.jors.2601319

Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, *6*(1), 80-91. https://doi.org/10.1287/mnsc.6.1.80

Derrac, J., García, S., Molina, D., & Herrera, F. (2011). A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation, 1*(1), 3-18. https://doi.org/10.1016/j.swevo.2011.02.002

Eberhart, R. C., & Shi, Y. (2001). Particle swarm optimization: developments, applications and resources, In *Congress on Evolutionary Computation, 1,* 81-86. https://doi.org/10.1109/CEC.2001.934374

Escobar, J. W., Linfati, R., Toth, P., & Baldoquin, M. (2014). A hybrid granular tabu search algorithm for the multi-depot vehicle routing problem. *Journal of Heuristics*, *20*(5), 1-27. https://doi.org/10.1007/s10732-014-9247-0

George, S., & Binu, S. (2018). Vehicle route optimisation using artificial bees colony algorithm and cuckoo search algorithm - a comparative study. *International Journal of Applied Engineering Research*, *13*(2), 953-959.

Ghobadi, A., Tavakkoli-Moghaddam, R., Fallah, M., & Kazemipoor, H. (2021). Multi-depot electric vehicle routing problem with fuzzy time windows and pickup/delivery constraints. *Journal of Applied Research on Industrial Engineering*, *8*(1), 1-18. https://doi.org/10.22105/jarie.2021.231764

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimisation and Machine Learning*, Reading, Mass.: Addison Wesley. https://doi.org/10.5555/534133

Golden, B. L., & Assad, A. A. (1988). *Vehicle Routing: Methods and Studies*. Elsevier Science, Amsterdam.

Goli, A., Aazami, A., & Jabbarzadeh, A. (2018). Accelerated cuckoo optimization algorithm for capacitated vehicle routing problem in competitive conditions. *International Journal of Artificial Intelligence*, *16*(1), 88-112.

Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris hawks optimization: algorithm and applications. *Future Generation Computer Systems*, *97*, 849-872. https://doi.org/10.1016/j.future.2019.02.028

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*, MIT Press. https://doi.org/10.7551/mitpress/1090.001.0001

Huang, L., Fu, Q., & Tong, N. (2023). An improved Harris hawks optimization algorithm and its application in grid map path planning. *Biomimetics, 8*(428). https://doi.org/10.3390/biomimetics8050428

Hussien, A. G., Abualigah, L., Zitar, R. A., Hashim, F. A., Amin, M., Saber, A., Almotairi, K., & Gandomi, A. H. (2022). Recent advances in Harris hawks optimization: a comparative study and applications. *Electronics*, *11*(12). https://doi.org/10.3390/electronics11121919

Juan, A. A., Pascual, I., Guimarans, D., & Barrios, B. (2014). Combining biased randomization with iterated local search for solving the multi-depot vehicle routing problem. *International Transactions in Operational Research*, *22*, 647-667. https://doi.org/10.1111/itor.12101

Kennedy, J., & Eberhart, R. C. (1995). Particle swarm optimization, In *IEEE Conference on Neural Networks, IV*, 1942-1948. https://doi.org/10.1109/ICNN.1995.488968

Kulworawanichpong, T., Puangdownreong, D., & Sujitjorn, S. (2004). Finite convergence of adaptive tabu search. *ASEAN Journal on Science and Technology for Development*, *21*(2&3), 103-115.

Li, H., & Lim, A. A. (2001). A metaheuristic for the pickup and delivery problem with time windows, In *International Conference on Tools with Artificial Intelligence*. https://doi.org/10.1109/ICTAI.2001.974461

Li, H., & Lim, A. A. (2003). A metaheuristic for the pickup and delivery problem with time windows. *International Journal of Artificial Intelligence Tools*, *12*(2), 160-170. https://doi.org/10.1142/S0218213003001186

Li, H., & Lim, A. A. (2008). Pickup and delivery problem with time windows instances, Available: https://www.sintef.no/projectweb/top/pdptw/li-lim-benchmark/. https://doi.org/10.1287/trsc.2022.1167

Lv, H., Liu, R., & Li, J. (2025). A multitasking ant system for multi-depot pick-up and delivery location routing problem with time window. *Complex & Intelligent Systems*, *11*(159). https://doi.org/10.1007/s40747-024-01750-3

Modares, A., Somhom, S., & Enkawa, T. (1999). A self-organizing neural network approach for multiple traveling salesman and vehicle routing problems. *International Transactions in Operational Research*, *6*(6), 591-606. https://doi.org/10.1111/j.1475-3995.1999.tb00175.x

Naccache, S., Côté, J-F., & Coelho, L. C. (2018). An adaptive large neighborhood search for the multi-pickup and delivery problem with time windows. *European Journal of Operational Research*, *269*(1). https://doi.org/10.1016/j.ejor.2018.01.035

Nagy, G., & Salhi, S. (2005). Heuristics algorithm for single and multiple depot vehicle routing problem with pickups and deliveries. *European Journal of Operational Research*, *162*, 126-141. https://doi.org/10.1016/j.ejor.2002.11.003

Polacek, M., Hartl, R., Doerner, K., & Reimann, M. (2004). A variable neighborhood search for the multidepot vehicle routing problem with time windows. *Journal of Heuristics*, *10*, 613-627. https://doi.org/10.1007/s10732-005-5432-5

Puangdownreong, D., Sujitjorn, S., & Kulworawanichpong, T. (2004). Convergence analysis of adaptive tabu search. *ScienceAsia Journal of the Science Society of Thailand*, *30*(2), 183-190. https://doi.org/10.2306/scienceasia1513-1874.2004.30.183

Ramos, T. R. P., Gomes, M. I., & Póvoa, A. P. B. (2020). Multi-depot vehicle routing problem: a comparative study of alternative formulations. *International Journal of Logistics Research and Applications*, *23*(2), 103-120. https://doi.org/10.1080/13675567.2019.1630374

Schaap, H., Schiffer, M., Schneider, M., Walther, Grit. (2022). A large neighborhood search for the vehicle routing problem with multiple time windows. *Transportation Science, 56*(2). https://doi.org/10.1287/trsc.2021.1120

Shi, Y., Lv, L., Hu, F., & Han, Q. (2020). A heuristic solution method for multi-depot vehicle routing-based waste collection problems. *Applied Science*, *10*(7). https://doi.org/10.3390/app10072403

Sombuntham, P., & Kachitvichayanukul, V. (2010). A particle swarm optimization algorithm for multidepot vehicle routing problem with pickup and delivery requests. In *International Multi-Conference of Engineers and Computer Scientists*, *3*.

Sombunthama, P., & Kachitvichyanukul, V. (2010). Multi-depot vehicle routing problem with pickup and delivery requests. *IAENG Transactions on Engineering Technologies*, *5*, 71-85. https://doi.org/10.1063/1.3510581

Sujitjorn, S., Kulworawanichpong, T., Puangdownreong, D., & Areerak, K-N. (2006). Adaptive tabu search and applications in engineering design. *Integrated Intelligent Systems for Engineering Design (Eds. X. F. Zha and R. J. Howlett)*, IOS Press, Netherland, 233-257.

Suwannarongsri, S. (2020). Application of modified flower pollination algorithm to multiple vehicle routing problems with time constraints. *WSEAS Transactions on Systems and Control*, *15*, 459-467. https://doi.org/10.37394/23203.2020.15.46

Suwannarongsri, S. (2021). Optimal solving multi-vehicle routing problems via parallel cuckoo search. *International Journal of Innovative Computing, Information and Control*, *17*(6), 1921-1935. https://doi.org/10.24507/ijicic.17.06.1921

Suwannarongsri, S. (2023). Optimal solving multi-depot vehicle routing problem by modern metaheuristic algorithm. *International Journal of Innovative Computing, Information and Control*, *19*(6), 1753-1767. https://doi.org/10.24507/ijicic.19.06.1753

Suwannarongsri, S. (2025). Application of parallel grey wolf optimizer for optimal solving multi-depot vehicle routing problem. *International Journal of Innovative Computing, Information and Control*, *21*(5), 1277-1296. https://doi.org/10.24507/ijicic.21.05.1277

Toklu, N. E., Gambardella, L. M., & Montemanni, R. (2014). A multiple ant colony system for a vehicle routing problem with time windows and uncertain travel times. *Journal of Traffic and Logistics Engineering*, *2*(1), 52-58. https://doi.org/10.12720/jtle.2.1.52-58

Tripathy, B. K., Maddikunta, P. K. R., Pham, Q-V., Gadekallu, T. R., Dev, K., Pandya, S., & El-Halawany, B. M. (2022). Harris hawk optimization: a survey on variants and applications. *Computational Intelligence and Neuroscience*, *2022*. https://doi.org/10.1155/2022/2218594

Wei, X., Xiao, Z., & Wang, Y. (2024). Solving the vehicle routing problem with time windows using modified rat swarm optimization algorithm based on large neighborhood search. *Mathematics, 12*(11). https://doi.org/10.3390/math12111702

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin, 1*(6), 80-83. https://doi.org/10.2307/3001968

Yang, Y., Sun, J., & Cui, J. (2020). Evolutionary game analysis on the choice of vehicle sharing travel modes. *ICIC Express Letters, Part B: Applications*, *11*(3), 289-295. https://doi.org/10.24507/icicelb.11.03.289

Zhou, W., & Li, Y. (2010). Improved genetic algorithm for multiple traveling salesman problem. In *International Asia Conference on Informatics in Control, Automation and Robotics*, *1*, 493-495. https://doi.org/10.1109/CAR.2010.5456787