Alweshah et al. Journal of Big Data (2022) 9:42
https://doi.org/10.1186/s40537-022-00593-4

Journal of Big Data

RESEARCH

Open Access

Vehicle routing problems based on Harris Hawks optimization

Mohammed Alweshah[1], Muder Almiani[2*], Nedaa Almansour[1], Saleh Al Khalaileh[1], Hamza Aldabbas[1], Waleed Alomoush[3] and Almahdi Alshareef[4]

*Correspondence: Almiani.M@gust.edu.kw
[2] Gulf University for Science and Technology, Mubarak Al-Abdullah, Kuwait
Full list of author information is available at the end of the article

Abstract

The vehicle routing problem (VRP) is one of the challenging problems in optimization and can be described as combinatorial optimization and NP-hard problem. Researchers have used many artificial intelligence techniques in order to try to solve this problem. Among these techniques, metaheuristic algorithms that can perform random search are the most promising because they can be used to find the right solution in the shortest possible time. Therefore, in this paper, the Harris hawks optimization (HHO) algorithm was used to attempt to solve the VRP. The algorithm was applied to 10 scenarios and the experimental results revealed that the HHO had a strong ability to check for and find the best route as compared to other metaheuristic algorithms, namely, simulated annealing and artificial bee colony optimization. The comparison was based on three criteria: minimum objective function obtained, minimum number of iterations required and satisfaction of capacity constraints. In all scenarios, the HHO showed clear superiority over the other methods.

Keywords: Vehicle routing problem, Harris Hawks Optimization, Metaheuristic, Optimization

Introduction

Urban expansion and the growth of cities in particular has led to an increase in factory activity and consequently an increase in various transportation services as companies in a variety of sectors strive to meet the economic and commercial needs as well as the service requirements of the urban population [1].This trend has generated an increase in logistical research aimed at ensuring that transportation services are both time-efficient and cost-effective in order to improve geographical coverage, speed of service and customer satisfaction [2–4].

One of the most important approaches that has been adopted to try to enhance the efficiency and growth of distribution systems is the vehicle routing problem (VRP), which was first proposed by Dantzig and Ramser in 1959 [5]. The VRP emerged from an existing optimization problem, namely, the traveling salesman problem (TSP) [6] and remains the most structured and researched transport model available. The VRP model attempts to discover how to deliver goods to customers by using a fleet consisting of a predefined number of vehicles.

The vehicle routing problem (VRP) is a catch-all term for a group of issues involving "vehicles" completing visits to "customers." The basic practical difficulty of distributing products to geographically dispersed consumers utilizing a number of trucks operating out of a single goods depot gives rise to these problems [7]. The challenge is to route the cars in such a way that the overall distance traveled (or time spent, money incurred, etc.) is kept to a minimum [8].

The VRP, also described as "vehicle scheduling," "vehicle dispatching," or simply "delivery," emerges frequently in real-world circumstances that are not directly connected to the delivery of commodities [9–11]. Collection of mail from mailboxes, pick-up of children by school buses, house-call tours by a doctor, preventive maintenance inspection tours, laundry delivery, and so on are all VRPs in which the "delivery" operation can be a collection, collection and/or delivery, or neither; and in which the "goods" and "vehicles" can take a variety of forms, some of which may not even be of a physical nature [12].

The key goal is to find the minimum-length route that starts and ends in a depot, taking into account a number of variants such as the time window, service time, and vehicle capacity among others [13]. The VRP is a combinatorial optimization and NP-hard problem [14, 15] and several algorithms have been proposed in order to solve it, yet it remains an incredibly important challenge in the field of combinational optimization [16]. The only small-scale VRP technology yet discovers the ultimate global solution [17]. The expertise of stream VRP algorithms is divided into exact and heuristics algorithms [18].

Classical heuristics for the VRP can be split into constructive heuristics and improvement heuristics [19]. The origin heuristics forever, progress from a solution to a best one in its neighborhoods [20].Heuristic methods execute a comparatively limited exploration of the search space and typically require a short computation time to output solutions that have good fitness [21]. Classification heuristics for the VRP include the savings algorithm, route-first cluster-second, cluster-first route-second, and insertion heuristics. Two kinds of improvement algorithms are suitable for the VRP, namely, intra-route heuristics and inter-route heuristics [22].

Metaheuristics are also widely used in optimization problems including the VRP [23–25].These smart algorithms are suitable for solving hard optimization problems that require a large amount of time because they are able to reach optimal solutions in a fast time [26, 27], which is largely due to their utilization of random search instead of full search when exploring the search space [28].

The major goal of developing metaheuristic algorithms is to address complicated optimization issues where previous optimization approaches have failed. These techniques are now widely regarded as some of the most practical approaches to solve a wide range of real-world issues. Metaheuristic could be used to solve any problem that can be expressed as a problem of optimization technique. Also, They are ease of implementation as well as they are efficiency and flexibility [29].

Many metaheuristic algorithms have been applied to optimization problems, such as the mine blast algorithm [30], monarch butterfly algorithm [31], and many others [32–36]. The metaheuristic algorithms that have been applied specifically to the VRP include artificial bee colony (ABC) [37], practical swarm optimization(PSO) [38], Ant Colony Optimization (ACO)[39],harmony search [40],genetic algorithm [41] and others

[42–49]. In line with the literature, this paper uses the Harris hawks optimization (HHO) algorithm to solve the VRP [50], as this algorithm has been shown to have scientific strength in the area of complex optimization problems [51].

The cooperative behavior and pursuit manner of Harris' hawks in nature, known as surprise pounce, is the fundamental inspiration for HHO. Several hawks work together to attack on a victim from different directions in an attempt to catch it off guard. Based on the dynamic nature of events and the prey's escape behaviors, Harris hawks can display a variety of pursuit patterns. To design an optimization method, this work mathematically duplicates such dynamic patterns and behaviors [50, 52, 53]. As an outcome, the Harris hawks optimization (HHO) algorithm was used to solve the VRP in this paper.

The rest of this paper is organized as follows: In "Related works", the related works on VRP are described. Then, in "Harris Hawks optimization", the HHO algorithm is discussed in detail. This is followed by a demonstration of the proposed method of using HHO for VRP in "Proposed method: HHO for VRP". After that, in "Experimental results", the experimental results are presented and analyzed. Finally, in "Conclusion" some conclusions are made and some possible directions for future improvements are suggested.

Related works

In recent times, metaheuristic algorithms have become an important part of the methods that have been developed to solve the VRP, largely because these algorithms have the ability to find optimal solutions due to their robust local and global search capabilities. Several previous studies have used these algorithms to solve the VRP, including Szeto, et al. [54], who presented a heuristic ABC for the capacitated VRP (CVRP). To improve the original heuristic for solving the CVRP, an improved version of the ABC heuristic was developed. Computing tests revealed that the improved heuristic ABC is capable of providing significantly stronger solutions for the CVRP than the original ABC and that it also consumes less computing energy.

On the other hand, Wang, et al. [55] addressed the periodic VRP with time window and service selection issue by building a heuristic algorithm based on improved ant colony optimization (IACO) and simulated annealing (SA), which they named multi-objective simulate annealingant colony optimization (MOSA-ACO). Also using SA, Rabbouch et al. [56] proposed an empirical-type SA to solve the CVRP. Their approach works incrementally by exploiting the last component of the worst practicable option. The results showed that their method has high search precision, optimal quicker entry, and the ability to classify all optimal rates while increasing the SA algorithm's convergence speed. In another SA-based research study, Ilhan [57] proposed using the population-based SA algorithm to solve the CVRP. Three operators were used to build the process: swap, inclusion and return. In fact, the algorithm used has been evaluated on 63 occasions. The research findings demonstrated that the proposed method has the ability to evaluate the best pathways for 23 instances, along with the presence of the percentage error value and the run time.

Another type of VRP was addressed in [58], where the author suggested an optimization algorithm for a hybrid ant colony and applied it to the multi-depot VRP (MDVRP). The suggested algorithm is based on a mixture of probabilistic techniques that are based on the normal actions of ants when foraging for food as well as on specific techniques that are based on the computational concepts of annealing. The experimental results showed the proposed algorithm is superior to other methods as it is able to produce the smallest mean error. Also, in [59], the authors recommended a way to minimize the MDVRP in terms of both distance traveled and time consumed by using inferential GAs. In their method, the proposed algorithm assigns each depot to its nearest customer and allows each customer to draw one customer at a time before all customers are allocated to drive. In addition, the algorithm uses Dijkstra's shortest path to evaluate each depot's closest customer. The findings showed that the suggested algorithm was calculated from distance and time of travel and thus confirmed its efficacy.

On the other hand, Sethanan and Jamrus [60] suggested the use of both an integer linear programming formulation and a novel hybrid differential evolution algorithm involving a fuzzy logic controller genetic operator to solve the backhaul and heterogeneous fleet routing problem for multi-trip vehicles. In their work, the objective was to minimize the cumulative cost associated with the distance traveled by using the traditional method of differential evolution and differential evolution with a selected genetic operator and real-settings fuzzy logic controller.

Meanwhile, in [61], the authors suggested a mathematical model to deal with the issue of the electric VRP (EVRP), taking into account the demands of electrical charging in the operations. To solve this optimization problem, the authors built a dedicated tabu search (TS) algorithm. The results of the improvement the VRP using this approach has obtained more reliable results than other methods. In another hybrid approach, Akbar and Aurachmana [62] also used a TS algorithm, in this case with a GA, in order to refine the CVRP with time windows (CVRPTW). One of the shortcomings of their work is its lack of animation by which to visualize the outcome and thereby encourage future studies. Nevertheless, their findings indicated that the proposed hybrid algorithm not only successfully minimizes the current route, but also estimates the optimal number of homogeneous fleets.

Based on the foregoing review of related works, it can be seen that several different metaheuristic algorithms have been used to solve the VRP and its variants. The results obtained by previous research studies showed that these smart algorithms can find the best routes for vehicles to reach customers in the least possible time. The success of these algorithms is due to the precision of their random search mechanism. and to the balance they achieve between the local and global search processes. It therefore seemed logical to investigate whether HHO would be able to find optimal solutions for the VRP.

Harris Hawks optimization

The HHO algorithm is a new swarm intelligence optimization algorithm that was proposed by Heidari et al. in 2019 [50]. The algorithm has been shown to perform efficiently in the optimization domain relative to other metaheuristic algorithms. Moreover, the algorithm can be extended to solve multiple types of optimization problems successfully, exhibiting a good level of performance [63]. Centered on the effects of such an algorithm in problems of intricacy optimization, the study uses it to increase the efficiency related to the VRP [50].

The basic notion behind the improved algorithm is derived from how Harris hawks hunt for food under desert conditions which are characterized by a scarcity of food [64]. To find their prey, Harris hawks engage in a variety of complex processes of mutual hunting and coordination to find, encircle, flush out and ultimately strike a possible target or prey. The HHO algorithm mimics the behavior of these hawks that monitor and catch their prey by means of a "surprise pounce", which is also known as the "seven kills" strategy [50].In this strategy, many hawks collaboratively target a single prey, such as a rabbit that is out in the open rather than hiding under cover of vegetation for example, by using different techniques and they ultimately converge on this prey [65].

The attack can be terminated promptly by catching the shocked target in just a few seconds. However, depending on the prey's escape capability and habits, the hawks may need to undertake several, short-length, swift dives that are likely to be in the vicinity of the prey for several minutes [51] before they are successful in executing the seven kills strategy. Harris hawks display a variety of types of chasing, which depends not only on the prey's escape habits, but also on a diverse range of environmental circumstances. For instance, in a swooping chase scenario, the most successful hawk (leader) swoops at the prey to capture it, but it can miss the prey (i.e., not find the target solution) because in essence the search and attack is carried out by a single member of the group. By increasing its vulnerability, Harris hawks proceed to locate an exhausted rabbit. In the end, when only one hawk, usually the most seasoned and strongest, can easily catch the exhausted rabbit and share it with the extant members, it cannot run away from the team besiege [66].

The HHO algorithm consists of three phases: (1) discovery or exploration (global search), (2) transformation from exploration to exploitation, and (3) exploitation (local search) [64]. For the intention of finding and investigating all aspects relevant to these propositions, the discovery phase reflects the change to many locations and a new venue. In addition, this phase illustrates a new technique for studying. The transformation from discovery to exploitation phase is regarded as crucial to the success of metaheuristic algorithms. In HHO, in order to transform the specified phases, the escape energy of the rabbit prey is used. The utilization process, on the other hand, reflects the installation and use of such sites and enables the available resources to be completely implemented [67].

Exploration phase

In this algorithm, the Harris hawks perch in certain positions frequently and wait to spot a target based on two techniques that are formulated in Eq. (1):

$$X(t+1) = \begin{cases} X(t+1) = X_{rand(t)} - r_1\left|X_{rand(t)} - 2r_2X(t)\right|, & q \geq 0.5 \\ X_{rabbit(t)} - X_{m(t)} - r_3(LB + r_1(UB - LB)), & q < 0.5 \end{cases} \tag{1}$$

where $X(t+1)$ is the hawks' location vector in the next iteration t, $X_{rabbit}(_{t})$ is the rabbit's position, $X(t)$ is the hawks' current location vector, r_1, r_2, r_3, r_4 and q are random numbers within (0,1) that are modified in each iteration, *LB* and *UB* represent the upper and lower variable borders, $X_{rand(t)}$ is a hawk randomly selected from the current population, and $X_{m(t)}$ is the mean variable position. The average position of the hawks is reached by using Eq. (2).

$$X_{m(t)} = \frac{1}{N}\sum\nolimits_{i=1}^{N}\left(X_{i(t)}\right), \tag{2}$$

where the position of each hawk in iteration t is indicated by $X_{i(t)}$ and N refers to the total number of hawks.

Transition from exploration to exploitation

In order to represent this step, the rabbit's energy is formulated as in Eq. (3):

$$E = 2E_0\left(1 - \frac{t}{T}\right), \tag{3}$$

where E denotes the fleeing energy of the rabbit, T is the cumulative number of iterations, and E_0 is the rabbit's initial energy state, and t is the current iteration.

Exploitation phase

In the final exploitation phase, the hawks employ four different types of besiege technique to capture their prey.

Soft besiege

The following rules in Eqs. (4) and (5) describe the soft besiege behavior of the Harris hawks:

$$X(t+1) = \Delta X(t) - E\left|JX_{rabbit(t)} - X(t)\right|, \tag{4}$$

$$\Delta X(t) = X_{rabbit(t)} - X(t), \tag{5}$$

where $X(t)$ is the variance between the rabbit's place vector and the specific position in iteration t during the escape process, r_5 is a random number within (0, 1), and $J = 2(1 - r_5)$ defines the rabbit's random leap ability. To mimic the nature of the rabbit's movements, the J value varies arbitrarily with each iteration.

Hard besiege

For a hard or intense besiege, the existing positions of the hawks are adjusted by Eq. (6):

$$X(t+1) = X_{rabbit(t)} - E|\Delta X(t)|. \tag{6}$$

Soft besiege with progressive rapid dives

When a soft besiege is accompanied by rapid dives, it is assumed that the hawks decide on adopting this technique based on the following rule in Eq. (7):

$$Y = X_{rabbit(t)} - E\left|JX_{rabbit(t)} - X(t)\right|. \tag{7}$$

It is also assumed that they use the following rule in Eq. (8) to dive at the prey based on Lévy flight-based patterns:

$$Z = Y + S \times LF(D), \tag{8}$$

where D is the issue aspect, S is a random vector of size 1 multiplied by D, and LF is the Lévy flight function that is determined using Eq. (9): where μ and σ are random values within (0, 1), and β is the default constant that is fixed to 1.5.

$$LF(x) = 0.01 \times \frac{\mu\sigma}{|\nu|^{\frac{1}{\beta}}}, \quad \sigma = \left[\frac{A(1+\beta) \times sin\left[\frac{\Pi\beta}{2}\right]}{\Gamma\left[\frac{1+\beta}{2}\right] \times \beta \times 2^{\left[\frac{\beta-1}{2}\right]}}\right]^{\frac{1}{\beta}} \tag{9}$$

The final technique that is used to upgrade the positions of the hawks that are in the process of undertaking a soft besiege with dives can then be carried out by Eq. (10):

$$X(t+1) = \begin{cases} Y, & f(Y) < f(X(t)) \\ Z, & f(Z) < f(X(t)) \end{cases} \tag{10}$$

where Y and Z are obtained by using Eqs. (7) and (8).

Hard besiege with progressive rapid dives

The final technique involves the combination of a hard besiege and rapid dives, as formulated in Eq. (11):

$$X(t+1) = \begin{cases} Y, & f(Y) < f(X(t)) \\ Z, & f(Z) < f(X(t)) \end{cases} \tag{11}$$

where Y and Z are obtained by using the new rules in Eqs. (12) and (13):

$$Y = X_{rabbit(t)} - E\left|JX_{rabbit(t)} - X(t)\right|. \tag{12}$$

$$Z = Y + S \times LF(D). \tag{13}$$

All of these steps of HHO are described in the following algorithm pseudocode [50]:

Pseudocode of HHO Algorithm
1- **Inputs:** Size N of the population and overall number of iterations (*N*).
2- **Outputs:** The positioning of rabbits as well as their fitness value.
3- **Initialize** Random population *Xi*, *i = 1, 2, . . . , n.* 4-While stopping position is not satisfied do: **5-** Determine the fitness values of hawks
6- Set $X_{rabbit(t)}$ as the location of rabbit with 'the best location'. **7-** for (every hawk (*Xi*)) do
8- Modify the initial energy E_0 as well as jump strength J
9- E_0=*2rand() - 1, J = 2 (1-rand)()*
10- Modify the by Eq. 3
11- if (| E | ≥ 1) then go to Exploration phase
12- Modify the location vector by Eq. 1
13- if (| E |< 1) then go to Exploitation phase
14- if (r ≥ 0.5 and | E | ≥ 0.5) then go to Soft besiege
15- Modify the location vector by Eq. 4
16- else if (r ≥ 0.5 and | E | < 0.5) then go to Hard besiege **17-** Modify the location vector by Eq. 6
18- else if (r < 0.5 and | E | ≥ 0.5) then go to Soft besiege
19- With quick radical dives
20- Modify the location vector using Eq. 10
21- else if (r < 0.5 and | E | < 0.5) then go to Hard besiege
22- With quick radical dives
23- Modify the location vector by Eq. 11
24- Return X_{rabbit}

Proposed method: HHO for VRP

In this paper, we suggest using HHO to construct a routing strategy by focusing on minimizing the number of vehicles in two steps:

Stage 1

The HHO global search functionality is used in the first step, with the HHO fitness function defined as:

$$f(\sigma) = \left(|\sigma|, \sum_{i=1}^{|\sigma|} c(\sigma_i) \right), \tag{14}$$

where |σ| is the number of routes in the σ routing strategy and c(σi) is the travel expense of the σi route ϵ σ.

In the first iteration, the rabbit randomly chooses a direction of travel. A fitness function is calculated when it hits its destination (e.g., the transport distance) and the artificial pheromones are evaluated and identified by the network. These artificial pheromones affect the next hawk's preference. In the subsequent iterations, the rabbits appear to take the shorter direction. In addition, the chance of better pathways being chosen can be improved by artificial pheromones.

The decrease in the number of vehicles is typically the complication, instead of the primary feature of the searching process, of reducing transport costs. In VRP-SPDTW, because reducing the number of vehicles and travel costs are both important targets, we concentrate on one of them in each process.

Stage 2

In the second stage, the local search of HHO minimizes the travel cost by using the routing plans that were generated in Stage 1 as the initial solution. The search in this stage continues until a certain stopping condition is met. Figure 1 shows the mechanism of the suggested method.

There are three conditions that must be taken into account when solving the routing problem: (1) the vehicle starts from the warehouse and returns to it, (2) the vehicle visits one customer only in one journey and then returns to the place of departure i.e., the warehouse) before going to another customer and (3) the customer may not request a delivery from more than one vehicle.

Furthermore, each vehicle must not exceed a specific load (capacity), which is defined as follows:

$$\text{Capacity} = \text{ceil}\left(\frac{N}{K}\right), \tag{15}$$

where N is the number of nodes (customers) and K is the number of vehicles.

As for addressing the issue of time complexity, the HHO's computational complexity depends primarily on three processes: initialization, fitness assessment, and hawk update [43]. The time complexity of the initialization process is O(N) for N hawks. While $O(T \times N) + 25\ O(T \times N \times D)$ is the computational complexity of the updating process, which consists of looking for the best location and updating all the hawks' location vectors, where T is the maximum number of iterations and D is the dimension of unique problems. Consequently, $O(N \times (T + TD + 1))$ is the computational complexity of HHO.

Experimental results

To provide a fair scientific examination, the experiments used the same work settings and situations. The proposed method was performed on a Notebook that had a Windows 10 operating system, An Intel® CoreTM i7-6006U processor operating at 2.00 GHz

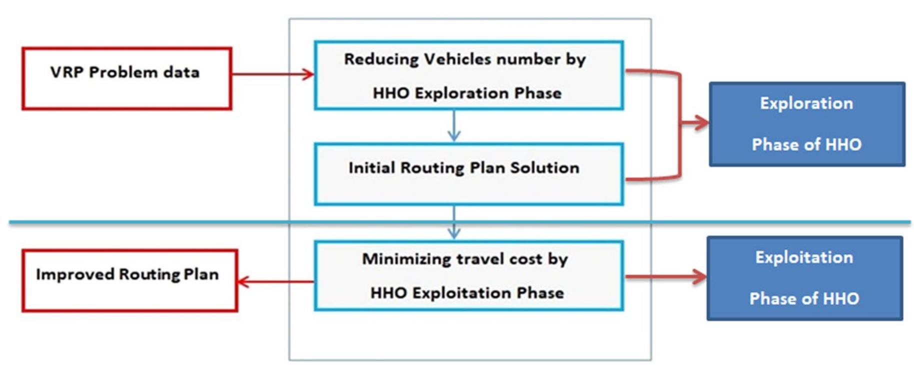

Fig. 1 Applying HHO in VRP

(four CPUs) and 2.0 GHz, with 8 GB of RAM, powers the software and its operation. The model was created using Matlab R2016a.

Since the input variables play an essential role in the consistency of the solutions. The input parameters were set in the experiments due to the obvious results as in [65]. In order for the outcomes of the experiment to be identical, the configurations of the algorithms are equivalent. The final set of parameters is specified in Table 1.

Application of HHO to 10 VRP scenarios

First, the HHO algorithm was applied to solve the VRP using 10 scenarios, where each scenario consisted of different numbers of customers and vehicles. In the first scenario, there were eight customers and three vehicles. These numbers were increased in each subsequent scenario to 70 customers and eight vehicles in the last scenario. This was done to make the problem more complex in order to measure the efficiency of the proposed approach using HHO in reaching the optimal solution for different sized problems. To assess the performance of HHO in each scenario, three criteria were calculated: (1) the minimum objective function obtained, (2) the minimum number of iterations required, and (3) the satisfaction of capacity constraints as explained in "Stage 2" Table 2 shows the results obtained by HHO for the 10 scenarios.

It can be seen from Table 2 that, in all 10 scenarios, HHO succeeded in reaching the optimum solution with a relatively low number of iterations. In addition, HHO was able to satisfy the capacity constraints in all scenarios.

Figure 2 depicts the routes taken by the vehicles in order to reach the customers according to the conditions of the VRP for each scenario. In the figure, the route of each vehicle is depicted using a different color and is the route used to reach each customer at the lowest possible cost.

As seen in Fig. 2, in all ten simulations. HHO was able to find the best solution with a very small number of iterations. Furthermore, in all instances, HHO was able to meet the capacity limits. As the number of customers increases, it takes a lot of roads that must be followed in order to reach all customers as soon as possible; therefore, the number of roads in the first simulation is different from what it is in the tenth.

Comparison of HHO against SA and ABC

In order to evaluate the efficacy of the proposed method, the outcomes of the proposed method were compared with those of two methods in the literature, namely, SA [48]

Table 1 Parameter settings

Parameter	Value
T: maximum number of iterations	200
N: population size	100
Lower bound (*LB*)	0
Upper bound (*UB*)	1
Number of hawks	10
Number of iterations	30
Model.etu (learning rote)	0.0001

Table 2 Results obtained by HHO for VRP

Simulation number	Minimum objective function obtained	Minimum number of iterations required
Sim. 1 8 customers 3 vehicles	220.1634	13
Sim. 2 10 customers 3 vehicles	269.8302	28
Sim. 3 14 customers 4 vehicles	219.2101	51
Sim. 4 20 customers 4 vehicles	297.457	63
Sim. 5 25 customers 5 vehicles	251.976	172
Sim. 6 30 customers 5 vehicles	316.1595	193
Sim. 7 40 customers 6 vehicles	319.0379	157
Sim. 8 50 customers 7 vehicles	298.1563	194
Sim. 9 60 customers 7 vehicles	401.5839	156
Sim. 10 70 customers 8 vehicles	423.0535	186

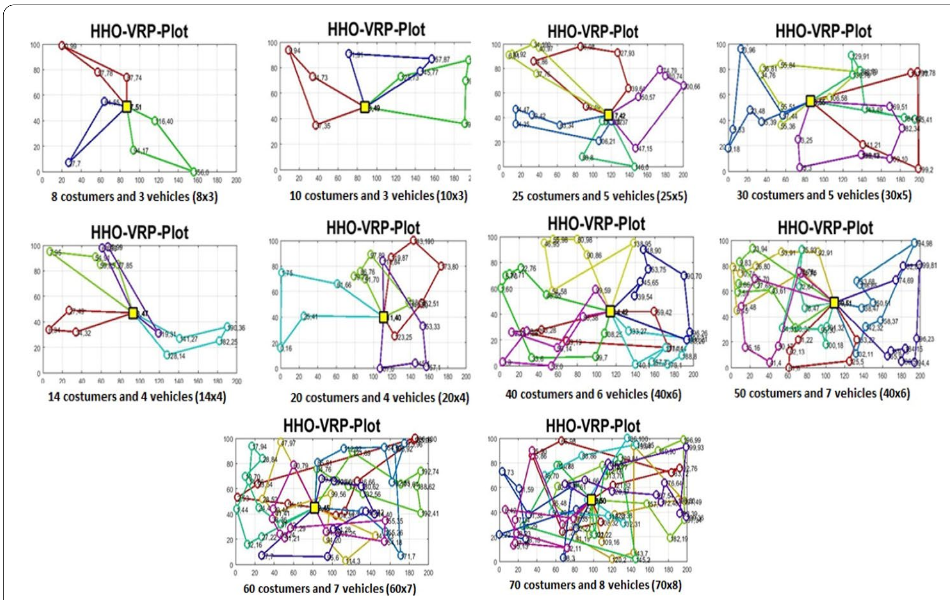

Fig. 2 VRP route maps for each simulation by HHO

and ABC [45]. SA and ABC are effective and largely used in many recent works in optimization field [68–72]. On the other hand, the comparison was made with two types of algorithms; single-based algorithm represented by SA, and population-based algorithm represented by ABC. The comparison with two different types of research method is very important to judge the proposed approach. The comparison was made by using the same parameter settings and criteria outlined above. Table 3 shows the outcomes of all three methods.

As can be seen from Table 3, in sim.1 with eight customers and three vehicles, all three methods obtained the same minimum objective function of 220.1634 and satisfied the capacity constraints. However, HHO did so more quickly in just 13 iterations. However, in sim. 2 with 10 customers and three cars, SA performed better than HHO and ABC with an objective function of 269.8302. However, again the least number of iterations was achieved by HHO with 28 iterations. All three methods met the capacity constraints of sim.2.

Table 3 Comparison of HHO, SA and ABC for VRP

Simulation number	Method	Minimum objective function obtained	Minimum number of iterations required
Sim. 1 8 customers 3 vehicles	HHO	220.1634	13
	SA	220.1634	51
	ABC	220.1634	61
Sim. 2 10 customers 3 vehicles	HHO	269.8302	28
	SA	293.9009	105
	ABC	284.9794	65
Sim. 3 14 customers 4 vehicles	HHO	219.2101	51
	SA	278.3779	230
	ABC	276.2006	144
Sim. 4 20 customers 4 vehicles	HHO	297.457	63
	SA	385.7173	264
	ABC	334.7739	231
Sim. 5 25 customers 5 vehicles	HHO	251.976	172
	SA	376.6583	248
	ABC	330.3308	224
Sim. 6 30 customers 5 vehicles	HHO	316.1595	193
	SA	419.6352	282
	ABC	354.6163	206
Sim. 7 40 customers 6 vehicles	HHO	319.0379	157
	SA	549.2129	294
	ABC	375.9213	280
Sim. 8 50 customers 7 vehicles	HHO	298.1563	194
	SA	626.9496	298
	ABC	377.2894	292
Sim. 9 60 customers 7 vehicles	HHO	401.5839	156
	SA	712.1566	300
	ABC	416.0673	250
Sim. 10 70 customers 8 vehicles	HHO	423.0535	186
	SA	694.4901	300
	ABC	442.3711	300

On the other hand, in sim.3 with 14 customers and four cars, HHO outperformed the other algorithms in terms of both the minimum objective function with 219.2101 and the fastest convergence with 51 iterations. Again all algorithms satisfied the capacity constraints. Similarly, in sim.4 with 20 customers and four vehicles, HHO outperformed the other algorithms in terms of objective function and number of iterations of 297.457 and 63, respectively. Likewise, in sim.5 and sim.6, HHO performed better than the other two methods achieving an objective function of 251.976 and 172 iterations and 316.1595 and 193, respectively.

Furthermore, HHO was able to achieve the lowest objective function and number of iterations in sim.7, sim.8, sim.9 and sim.10. In sim.7, with 40 customers and six cars, it achieved an objective function of 319.0379 in 157 iterations. In sim.8 with 50 customers and seven vehicles, it had an objective function of 298.1563 and convergence speed of 194 iterations. In sim.9 with 60 customers and seven vehicles, its objective function was 401.5839 and its number of iterations was 156.

Importantly, in sim.10 that had the highest level of complexity with 70 customers and eight vehicles, the lowest objective function was also achieved by HHO with 423.0535. The proposed method also performed the best in terms of convergence speed with 186 iterations.

Furthermore, a T-test was performed to compare the effectiveness of the HHO technique. The provided methodologies are utilized to calculate the findings statistics, which are based on the simulation's minimum objective function. Table 4 shows the results of a T-test on p-values and minimum objective function by the HHO, SA, and ABC, along with a 95 percent confidence interval (alpha value = 0.05).

As shown in Table 4,the HHO is more efficient than the SA and ABC, where the total number of p-values used in all simulations is less than 0.0001.As well as, the HHO has the least standard deviation and standard error rate in all simulations expect simulation 5. These findings suggest that the HHO is effective in resolving VRP.

In sum, in all ten scenarios, all three methods, HHO, SA and ABC, were able to satisfy the capacity constraints. However, HHO was clearly superior to the other two methods in terms of the minimum objective function obtained and the minimum number of iterations required. This indicates that HHO has the ability to achieve a balance between local and global search to reach more accurate solutions in a fast time. Table 5 provides more details on the number of iterations required in each simulation to reach the optimal solution compared to the other two methods.

From Table 5, the strength of HHO in every simulation is readily apparent. The use of HHO is advantageous whether the size of the problem is minimal, moderate or large, as it is able to find an optimal solution in the least number of iterations as compared to the other compared methods. Harris hawk optimization is not only able to find an optimal solution, it can do so quickly. This is due to its architecture, which achieves a better balance between exploration and exploitation, and it can transition smoothly between these two types of search. The number of iterations, i.e., convergence speed, is an important factor in solving complex optimization problems such as the VRP in a timely and cost-effective manner. Hence these results show that HHO has great potential for solving the VRP.

Table 4 T-test results for HHO, SA and ABC

	Scenario	Method	Mean	Std. deviation	Std. error mean	P_value
1	8 customers 3 vehicles	HHO	220.1634	1.229605574158	0.549896329865	0.00
		SA	220.1634	2.011608776079	0.899618793489	0.00
		ABC	220.1634	2.578602309004	1.153186009974	0.00
2	10 customers 3 vehicles	HHO	269.8302	0.836660026534	0.374165738677	0.00
		SA	293.9009	3.209361307176	1.435270009441	0.00
		ABC	284.9794	2.683281573000	1.200000000000	0.00
3	14 customers 4 vehicles	HHO	219.2101	0.547722557505	0.244948974278	0.00
		SA	278.3779	3.807886552932	1.702938636593	0.00
		ABC	276.2006	3.633180424917	1.624807680927	0.00
4	20 customers 4 vehicles	HHO	297.457	0.894427190999	0.400000000000	0.00
		SA	385.7173	4.438468204234	1.984943324128	0.00
		ABC	334.7739	6.188699378706	2.76767050062	0.00
5	25 customers 5 vehicles	HHO	251.976	5.630275304104	2.517935662403	0.00
		SA	376.6583	5.549774770205	2.481934729198	0.00
		ABC	330.3308	3.114482300479	1.392838827718	0.00
6	30 customers 5 vehicles	HHO	316.1595	0.547722557505	0.244948974278	0.00
		SA	419.6352	7.155417527999	3.200000000000	0.00
		ABC	354.6163	6.496152707565	2.905167809267	0.00
7	40 customers 6 vehicles	HHO	319.0379	2.607680962081	1.166190378969	0.00
		SA	549.2129	7.463243262818	3.337663853656	0.00
		ABC	375.9213	3.563705936241	1.593737745051	0.00
8	50 customers 7 vehicles	HHO	298.1563	1.949358868962	0.871779788708	0.00
		SA	626.9496	4.669047011972	2.088061301782	0.00
		ABC	377.2894	4.658325879541	2.083266665600	0.00
9	60 customers 7 vehicles	HHO	401.5839	0.894427190999	0.400000000000	0.00
		SA	712.1566	1.483239697419	0.663324958071	0.00
		ABC	416.0673	5.099019513593	2.2803508501 98	0.00
10	70 customers 8 vehicles	HHO	423.0535	1.303840481041	0.583095189484	0.00
		SA	694.4901	4.207136793593	1.881488772223	0.00
		ABC	442.3711	7.127411872482	3.187475490102	0.00

Conclusion

The VRP is an NP hard real-world problem, which many researches are still trying to provide the best solutions to solve it. In this paper, we proposed a model using HHO algorithm to solve the VRP. The performance of HHO was assessed according to three criteria: the minimum objective function obtained, the minimum number of iterations required, and the satisfaction of capacity constraints. Compared with two well-known methods, namely, SA and ABC, the results showed that HHO was clearly superior in terms of both the minimum objective function obtained and the minimum number of iterations required. This is likely due to the advanced exploitative capabilities of HHO as well as its ability to achieve better integration between the local and global search processes.

On the other hand, HHO like other metaheuristic algorithms, cannot prove optimality and it cannot probably reduce the search space, as well as it is hard to guarantee the consistency of optimization outcomes acquired with the same initial condition settings. In the future, to achieve better results, HHO could be employed in a hybrid model with

Table 5 Number of iterations needed to get best solution

Simulation number	Number of iterations needed to get best solution		
	HHO	SA	ABC
Sim. 1 8 customers 3 vehicles	15	50	70
Sim. 2 10 customers 3 vehicles	30	100	100
Sim. 3 14 customers 4 vehicles	50	80	125
Sim. 4 20 customers 4 vehicles	70	120	90
Sim. 5 25 customers 5 vehicles	140	250	200
Sim. 6 30 customers 5 vehicles	190	290	250
Sim. 7 40 customers 6 vehicles	150	280	240
Sim. 8 50 customers 7 vehicles	145	280	260
Sim. 9 60 customers 7 vehicles	150	280	250
Sim. 10 70 customers 8 vehicles	180	280	300

another metaheuristic algorithm, or its global search mechanism could be modified through the use of several methods such as a crossover operator.

Acknowledgements
This work is supported by the Gulf University for Science and Technology, Kuwait.

Author contributions
We attest that all authors contributed significantly to the creation of this manuscript, each having fulfilled criteria. We confirm that the manuscript has been read and approved by all named authors. We confirm that the order of authors listed in the manuscript has been approved by all. All authors read and approved the final manuscript.

Funding
No funding was received for this work.

Availability of data and materials
The data that support the findings of this study are available on request.

Declarations

Ethics approval and consent to participate
We confirm that any aspect of the work covered in this manuscript that has involved human patients has been conducted with the ethical approval of all relevant bodies and that such approvals are acknowledged within the manuscript.

Consent for publication
We give my consent for the publication of identifiable details, which can include photograph(s) and/or videos and/or case history and/or details within the text ("Material") to be published in the above Journal and Article. I confirm that I have seen and been given the opportunity to read both the Material and the Article to be published by Journal of big data.

Competing interests

The author(s) declare(s) that they have no competing interests. We wish to confirm that there are no known conflicts of interest associated with this publication.

Author details

[1]Prince Abdullah Bin Ghazi Faculty of Information and Communication Technology, Al-Balqa Applied University, Al-Salt, Jordan. [2]Gulf University for Science and Technology, Mubarak Al-Abdullah, Kuwait. [3]School of Information Technology, Skyline University College, P.O. Box 1797, Sharjah, United Arab Emirates. [4]Faculti of Information Technology, Sebha University, Sebha, Libya.

Received: 14 October 2021 Accepted: 28 March 2022
Published online: 25 April 2022

References

1. Nazari M, Oroojlooy A, Snyder L, Takác M. Reinforcement learning for solving the vehicle routing problem. In: Advances in Neural Information Processing Systems, 2018, pp. 9839–9849.
2. Braekers K, Ramaekers K, Van Nieuwenhuyse I. The vehicle routing problem: state of the art classification and review. Comput Ind Eng. 2016;99:300–13.
3. Mohammed MA, AbdGhani MK, Hamed RI, Mostafa SA, Ahmad MS, Ibrahim DA. Solving vehicle routing problem by using improved genetic algorithm for optimal solution. J Comput Sci. 2017;21:255–62.
4. Zhao P, Luo W, Han X. Time-dependent and bi-objective vehicle routing problem with time windows. Adv Prod Eng Manage. 2019;14:201–12.
5. Marinakis Y, Migdalas A. Annotated bibliography in vehicle routing. Oper Res Int J. 2007;7:27–46.
6. Berghida M, Boukra A. Quantum inspired algorithm for a VRP with heterogeneous fleet mixed backhauls and time windows. Int J Appl Metaheuristic Comput. 2016;7:18–38.
7. Eksioglu B, Vural AV, Reisman A. The vehicle routing problem: a taxonomic review. Comput Ind Eng. 2009;57:1472–83.
8. Toth P, Vigo D. The vehicle routing problem. SIAM, 2002.
9. Bellman RP, Udeanu SR, Hammer PL, Fouquet ET. *Méthodes booléennes en recherche opérationnelle*: Dunod, 1970.
10. Himmelblau D. Some practical experiences in applying nonlinear programming to CAD. Comput Aided Des. 1981;13:317–26.
11. Christofides N. The vehicle routing problem, *Revue française d'automatique, informatique, recherche opérationnelle*. Recherche opérationnelle. 1976;10:55–70.
12.]12[Beale E, Small R. Mixed integer programming by a branch and bound technique. In: Proceedings of the IFIP Congress, 1965, pp. 450–451.
13. Zeng HY. Improved particle swarm optimization based on Tabu search for VRP. J Appl Sci Eng Innov. 2019;6:99–103.
14. Gayialis SP, Konstantakopoulos GD, Tatsiopoulos IP. Vehicle routing problem for urban freight transportation: a review of the recent literature. In: Operational research in the digital Era—ICT challenges. Berlin: Springer; 2009. p. 89–104.
15. Soleimani H, Chaharlang Y, Ghaderi H. Collection and distribution of returnedremanufactured products in a vehicle routing problem with pickup and delivery considering sustainable and green criteria. J Clean Prod. 2018;172:960–70.
16. Iswari T, Asih AMS. Comparing genetic algorithm and particle swarm optimization for solving capacitated vehicle routing problem. In: IOP Conference Series: Materials Science and Engineering, 2018, p. 012004.
17. Saeheaw T, Charoenchai N. A comparative study among different parallel hybrid artificial intelligent approaches to solve the capacitated vehicle routing problem. Int J Bio-Inspired Comput. 2018;11:171–91.
18.]18[Raeesi R. Mathematical models and solution algorithms for the vehicle routing problem with environmental considerations. Lancaster University, 2019.
19. Stützle T, López-Ibáñez M. Automated design of metaheuristic algorithms. In: Handbook of metaheuristics. Berlin: Springer; 2019. p. 541–79.
20. Pei J, Mladenović N, Urošević D, Brimberg J, Liu X. Solving the traveling repairman problem with profits: a Novel variable neighborhood search approach. Inf Sci. 2020;507:108–23.
21. Matl P, Hartl RF, Vidal T. Leveraging single-objective heuristics to solve bi-objective problems: Heuristic box splitting and its application to vehicle routing. Networks. 2019;73:382–400.
22. Khabou A. Transportation optimization for the collection of end-of-life vehicles. École de technologie supérieure, 2019.
23. Rasku J, Musliu N, Kärkkäinen T. On automatic algorithm configuration of vehicle routing problem solvers. J Vehicle Routing Alg. 2019;2:1–22.
24. Albashish D, Sahran S, Abdullah A, Adam A, Alweshah M. A hierarchical classifier for multiclass prostate histopathology image Gleason grading. J Inform Commun Technol. 2018;17:323–46.
25. Alweshah M, Qadoura MA, Hammouri AI, Azmi MS, AlKhalaileh S. Flower pollination algorithm for solving classification problems. Int J Adv Soft Comput Appl. 2020;12.
26. Alweshah M, Ramadan E, Ryalat MH, Almi'ani M, Hammouri AI. Water evaporation algorithm with probabilistic neural network for solving classification problems. Jordanian J Comput Inform Technol (JJCIT). 2020;6.
27. Alweshah M, Rababa L, Ryalat MH, Al Momani A, Ababneh MF. African buffalo algorithm: Training the probabilistic neural network to solve classification problems. J King Saud Univ-Comput Inform Sci. 2020.
28. Alweshah M. Construction biogeography-based optimization algorithm for solving classification problems. Neural Comput Appl. 2019;31:5679–88.
29. Alweshah M, Alkhalaileh S, Al-Betar MA, Bakar AA. Coronavirus herd immunity optimizer with greedy crossover for feature selection in medical diagnosis. Knowl-Based Syst. 2022;235:107629.

30. Alweshah M, Alkhalaileh S, Albashish D, Mafarja M, Bsoul Q, Dorgham O. A hybrid mine blast algorithm for feature selection problems. Soft Comput. 2020; 1–18.
31. Alweshah M, Al Khalaileh S, Gupta BB, Almomani A, Hammouri AI, Al-Betar MA. The monarch butterfly optimization algorithm for solving feature selection problems. Neural Comput Appl. 2020; 1–15.
32. Tongur V, Hacibeyoglu M, Ulker E. Solving a big-scaled hospital facility layout problem with meta-heuristics algorithms. Eng Sci Technol Int J. 2020;23:951–9.
33. Almufti SM. Historical survey on metaheuristics algorithms. Int J Sci World. 2019;7:1.
34. Cuevas E, Espejo EB, Enríquez AC. Metaheuristics algorithms in power systems, vol. 822. Berlin: Springer; 2019.
35. Öztop H, Tasgetiren MF, Eliiyi DT, Pan Q-K. Metaheuristic algorithms for the hybrid flowshop scheduling problem. Comput Oper Res. 2019;111:177–96.
36. Almomani A, Alweshah M, Al S. Metaheuristic algorithms-based feature selection approach for intrusion detection. In: machine learning for computer and cyber security: principle, algorithms, and practices. 2019; p. 184.
37. Shao K, Zhou K, Qiu J, Zhao J. ABC algorithm for VRP. In: Bio-inspired computing theories and applications. Berlin: Springer; 2014. p. 370–3.
38. He Y, Wen J, Huang M. Study on emergency relief VRP based on clustering and PSO. In: 2015 11th International Conference on Computational Intelligence and Security (CIS), 2015, pp. 43–47.
39. Diego FJ, Gómez EM, Ortega-Mier M, García-Sánchez Á. Parallel CUDA architecture for solving de VRP with ACO. In: Industrial engineering: innovative networks. Berlin: Springer; 2012. p. 385–93.
40. Yassen ET, Ayob M, Nazri MZA, Sabar NR. Meta-harmony search algorithm for the vehicle routing problem with time windows. Inf Sci. 2015;325:140–58.
41. Abedinzadeh S, Ghoroghi A, Erfanian HR. Application of hybrid GA-SA heuristic for green location routing problem with simultaneous pickup and delivery.
42. Golden BL, Wasil EA, Kelly JP, Chao I-M. The impact of metaheuristics on solving the vehicle routing problem: algorithms, problem sets, and computational results. In: Fleet management and logistics. Berlin: Springer; 1998. p. 33–56.
43. Bräysy O, Gendreau M. Vehicle routing problem with time windows, Part II: Metaheuristics. Transp Sci. 2005;39:119–39.
44. Van Breedam A. Comparing descent heuristics and metaheuristics for the vehicle routing problem. Comput Oper Res. 2001;28:289–315.
45. Bianchi L, Birattari M, Chiarandini M, Manfrin M, Mastrolilli M, Paquete L, Rossi-Doria O, Schiavinotto T. Hybrid metaheuristics for the vehicle routing problem with stochastic demands. J Mathe Model Alg. 2006;5:91–110.
46. Song M-X, Li J-Q, Han Y-Q, Han Y-Y, Liu L-L, Sun Q. "Metaheuristics for solving the vehicle routing problem with the time windows and energy consumption in cold chain logistics. Appl Soft Comput. 2020;95:106561.
47. Dixit A, Mishra A, Shukla A. Vehicle routing problem with time windows using metaheuristic algorithms: a survey. In: Harmony search and nature inspired optimization algorithms. Berlin: Springer; 2019. p. 539–46.
48. Gutierrez-Rodríguez AE, Conant-Pablos SE, Ortiz-Bayliss JC, Terashima-Marín H. Selecting meta-heuristics for solving vehicle routing problems with time windows via metalearning. Expert Syst Appl. 2019;118:470–81.
49. Elshaer R, Awad H. A taxonomic review of metaheuristic algorithms for solving the vehicle routing problem and its variants. Comput Indus Eng. 2020;140:106242.
50. Heidari AA, Mirjalili S, Faris H, Aljarah I, Mafarja M, Chen H. Harris hawks optimization: algorithm and applications. Futur Gener Comput Syst. 2019;97:849–72.
51. Yıldız BS, Yıldız AR. The Harris hawks optimization algorithm, salp swarm algorithm, grasshopper optimization algorithm and dragonfly algorithm for structural design optimization of vehicle components. Materials Testing. 2019;61:744–8.
52. Moayedi H, Osouli A, Nguyen H, Rashid ASA. A novel Harris hawks' optimization and k-fold cross-validation predicting slope stability. Eng Comput. 2021;37:369–79.
53. Chen H, Heidari AA, Chen H, Wang M, Pan Z, Gandomi AH. Multi-population differential evolution-assisted Harris hawks optimization: framework and case studies. Futur Gener Comput Syst. 2020;111:175–98.
54. Szeto WY, Wu Y, Ho SC. An artificial bee colony algorithm for the capacitated vehicle routing problem. Eur J Oper Res. 2011;215:126–35.
55. Wang Y, Wang L, Chen G, Cai Z, Zhou Y, Xing L. An improved ant colony optimization algorithm to the periodic vehicle routing problem with time window and service choice. Swarm Evol Comput. 2020;100675.
56. Rabbouch B, Saâdaoui F, Mraihi R. Empirical-type simulated annealing for solving the capacitated vehicle routing problem. J Exp Theor Artif Intell. 2020;32:437–52.
57. Ilhan I. A population based simulated annealing algorithm for capacitated vehicle routing problem. Turk J Electr Eng Comput Sci. 2020;28:12171235.
58. Stodola P. Hybrid ant colony optimization algorithm applied to the multi-depot vehicle routing problem. Nat Comput. 2020; 1–13.
59. Muazu AA, Nura A. Efficient assignment algorithms for multi depot vehicle routing problem using genetic algorithm. Ilorin J Comput Sci Inform Technol. 2020;3:1–10.
60. Sethanan K, Jamrus T. Hybrid differential evolution algorithm and genetic operator for multi-trip vehicle routing problem with backhauls and heterogeneous fleet in the beverage logistics industry. Comput Indus Eng. 2020;106571.
61. Wang Q, Peng S, Liu S. Optimization of electric vehicle routing problem using tabu search. In: 2020 Chinese Control And Decision Conference (CCDC), 2020, pp. 2220–2224.
62. Akbar M, Aurachmana R. Hybrid genetic–tabu search algorithm to optimize the route for capacitated vehicle routing problem with time window. Int J Ind Optim. 2020;1:15.
63. Chen H, Jiao S, Wang M, Heidari AA, Zhao X. Parameters identification of photovoltaic cells and modules using diversification-enriched Harris hawks optimization with chaotic drifts. J Cleaner Product. 2020;244:118778.
64. Chen H, Heidari AA, Chen H, Wang M, Pan Z, Gandomi AH. Multi-population differential evolution-assisted Harris hawks optimization: framework and case studies. Future Gen Comput Syst. 2020.

65. Bui DT, Moayedi H, Kalantar B, Osouli A, Pradhan B, Nguyen H, Rashid ASA. A novel swarm intelligence—Harris hawks optimization for spatial assessment of landslide susceptibility. Sensors. 2019;19:3590.
66. Kurtuluş E, Yıldız AR, Sait SM, Bureerat S. A novel hybrid Harris hawks-simulated annealing algorithm and RBF-based metamodel for design optimization of highway guardrails. Materials Testing. 2020;62:251–60.
67. Abbasi A, Firouzi B, Sendur P. On the application of Harris hawks optimization (HHO) algorithm to the design of microchannel heat sinks. Eng Comput. 2019;1–20.
68. Thirugnanasambandam K, Rajeswari M, Bhattacharyya D, Kim J-Y. Directed Artificial Bee Colony algorithm with revamped search strategy to solve global numerical optimization problems. Autom Softw Eng. 2022;29:1–31.
69. Zhang X, Tang L, Chu S-C, Weng S, Pan J-S. Hybrid optimization algorithm based on QUATRE and ABC algorithms. In: Advances in smart vehicular technology transportation communication and applications. Berlin: Springer; 2020. p. 187–97.
70. Jia D. A culture-based artificial bee colony algorithm for optimization in dynamic environments. J Adv Comput Intell Intell Informat. 2022;26:23–7.
71. Gao Q, Xu H, Li A. The analysis of commodity demand predication in supply chain network based on particle swarm optimization algorithm. J Comput Appl Math. 2022;400:113760.
72. Peters E, Shyamsundar P, Li AC, Perdue G. Noise-aware qubit assignment on NISQ hardware using simulated annealing and Loschmidt Echoes. arXiv preprint arXiv:2201.00445, 2022.

Publisher's Note