International Journal of Machine Learning and Cybernetics
https://doi.org/10.1007/s13042-023-02087-y

ORIGINAL ARTICLE

A hospitalization mechanism based immune plasma algorithm for path planning of unmanned aerial vehicles

Selcuk Aslan[1]

Received: 24 May 2023 / Accepted: 22 December 2023

Abstract
Unmanned aerial vehicles (UAVs) and their specialized variants known as unmanned combat aerial vehicles (UCAVs) have triggered a profound change in the well-known military concepts and researchers from different disciplines tried to solve challenging problems of the mentioned vehicles. Path planning is one of these challenging problems about the UAV or UCAV systems and should be solved carefully by considering some optimization requirements defined for the enemy threats, fuel or battery usage, kinematic limitations on the turning and climbing angles in order to further improving the task success and safety of autonomous flight. Immune plasma algorithm (IP algorithm or IPA) modeling the details of a medical method gained popularity with the COVID-19 pandemic has been introduced recently and showed promising performance on solving a set of engineering problems. However, IPA requires setting the control parameters appropriately for maintaining a balance between the exploration and exploitation characteristics and does not design the particular treatment and hospitalization procedures by taking into account the implementation simplicity. In this study, IP algorithm was supported with a newly designed and realistic hospitalization mechanism that manages when an infected population member enters and discharges from the hospital. Moreover, the existing treatment schema of the algorithm was changed completely for improving the efficiency of the plasma transfer operations and removing the necessity of IPA specific control parameters and then a novel path planner called hospital IPA (hospIPA) was presented. For investigating the performance of hospIPA on solving path planning problem, a set of detailed experiments was carried out over twenty test cases belonging to both two and three-dimensional battlefield environments. The paths calculated by hospIPA were also compared with the calculated paths of other fourteen meta-heuristic based path planners. Comparative studies proved that the hospitalization mechanism making an exact discrimination between the poor and qualified solutions and modified treatment schema collecting the plasma being transferred by guiding the best solution give a tremendous contribution and allow hospIPA to obtain more safe and robust paths than other meta-heuristics for almost all test cases.

Keywords Meta-heuristics · IP algorithm · Hospitalization · Unmanned aerial vehicles · Path planning

1 Introduction

The recent advances on the microcomputers, remote sensing methods, communication technologies and smart munition production approaches started a revolution for the design and usage concepts of unmanned aerial vehicles (UAVs) and unmanned combat aerial vehicles (UCAVs). Even though the developed countries have strong air forces containing the trained pilots, fighter jets, bombers and helicopters, they also allocate defence budgets for producing UAV or UCAV systems or increasing their task performances and capabilities by trying to solve complex problems about these vehicles [1]. In order to increase the task performance of a UAV or UCAV system, the first and foremost problem that should be solved optimally is the path planning. For planning the optimal or near optimal paths before the flight of a UAV or UCAV being operated, some classical techniques including Artificial Potential Field (APF), Probabilistic Road Map (PRM), Rapid-Exploring Random Trees (RRT) and well-known graph based methods such as A*, D* and Voronoi diagram were successfully experimented [2]. However, all of these path planners have difficulties about trapping local minimum solutions and require detailed information for the

✉ Selcuk Aslan
selcukaslan@erciyes.edu.tr

[1] Department of Aeronautical Engineering, Erciyes University, Kayseri, Turkey

Published online: 22 January 2024

battlefield in a map format [2]. Because of the mentioned limitations of the classical techniques, researchers tried to find alternative ways and discovered the potential of meta-heuristic algorithms to plan UAV paths or solve other complex engineering problems [3–5].

The meta-heuristic algorithms introduced and used with the standard or modified implementations for solving path planning problem of UAV or UCAV systems inspire from the intelligent behaviors of different species such as birds, ants, bees, butterflies, moths and wolves or try to model evolutionary mechanisms including natural selection, mutation and crossover or guide physical or chemical phenomenons such as gravitation, explosion, burning and annealing. However, the new coronavirus first seen in Wuhan, China at the beginning of 2020 and caused a global health crisis altered the main focus of researchers from computer and information sciences and they investigated how a medical method or treatment approach used for the patients infected with the coronavirus can be referenced to design and develop modern meta-heuristic techniques [6]. Immune Plasma algorithm (IP algorithm or IPA) is the first meta-heuristic directly utilizing from the fundamental steps of a medical method called immune or convalescent plasma treatment as the given name implies [7]. The promising performance of the standard IPA has been validated recently on big data optimization [8], radio channel assignment [9], wireless sensor deployment [10], neural network training [11], in addition to the UAV or UCAV path planning [12]. Even though the standard implementation of the IPA shows promising performance on different optimization problems, it still requires configuring the control parameters responsible for determining the number of donors and receivers subtly or modeling and then integrating more detailed treatment procedures. In this study,

- A new variant of the IPA called the hospital IPA (hospIPA) was proposed.
- The hospIPA integrated a newly designed and realistic hospitalization mechanism into the workflow of the IPA for controlling when an infected population member will enter and discharge from the hospital.
- The plasma collection and transfer schema was redesigned for hospIPA by aiming at increasing the treatment efficiency of the hospitalized individual or individuals.
- Moreover, the proposed plasma collection and transfer schema removed the necessity of the IPA specific control parameters determining how many individuals will be receivers and how many individuals will be plasma donors.

The path planning performance of hospIPA was investigated by using twenty test cases in total belonging to both two and three-dimensional battlefield environments. The paths obtained by the hospIPA were compared with the calculated paths of a set of meta-heuristics including IPA, Genetic algorithm (GA) and Particle Swarm Optimization algorithm (PSO) based version of GA called GAPSO, Moth Flame Optimization (MFO), Salp Swarm algorithm (SSA), Pathfinder algorithm (PFA), Stain Bowerbird Optimization (SBO), Sine-Cosine algorithm (SCA), Grey Wolf Optimizer (GWO) and its hybridization with the Symbiotic Organism Search (SOS) known as HSGWO-MSOS, Artificial Ecosystem Optimizer (AEO) and adaptive neighborhood-based search enhanced AEO (NSEAEO), a strong implementation of the Teaching-Learning Based Optimization (TLBO) algorithm and finally the comprehensively improved PSO for short CIPSO. Comparative studies between hospIPA and other meta-heuristic based planners showed that hospIPA is capable of calculating more safe, fuel efficient and flyable paths for the vast majority of the test cases. While the newly designed hospitalization approach allows hospIPA to discriminate the poor solutions from the remaining part of the population and helps exploring the vicinity of qualified solutions more steadily, the proposed treatment schema significantly improves the exploitation performance and gives a chance to update a poor solution with a candidate better than the best solution found until the current cycle. The rest of the paper is organized as follows: Mathematical model of the path planning problem is explained in Sect. 2. Fundamental steps of the IP algorithm are given in Sect. 3. Details of the newly proposed hospitalization and treatment procedures are mentioned in Sect. 4. Section 5 is devoted to the experimental and comparative studies. Finally, in Sect. 6, some final remarks and future works about the IPA based path planners are presented.

1.1 Related works

One of the first studies that illustrates how a meta-heuristic can plan optimal flight path of a UCAV was presented by Duan et al. over the Ant Colony Optimization (ACO) algorithm [13]. In another study, Duan et al. introduced a hybrid approach by combining ACO and Differential Evolution (DE) algorithms and experimented their path planner for a single UCAV being operated in a three-dimensional environment [14]. Ma and Lei related the values being assigned to the inertia weight of PSO algorithm with the second order oscillation links and second-order oscillating PSO (SOPSO) was presented [15]. Xu et al. designed a new Artificial Bee Colony (ABC) algorithm in which each employed forager searches food sources within the neighborhood of the current best solution by using chaotic random numbers and illustrated the effectiveness of new path planner against standard ABC [16]. For a more clear discrimination between the food sources of ABC algorithm, Zhang et al. scaled the raw fitness values in their path planning technique [17]. Zhang et al. also introduced Fitness-scaling Adaptive Chaotic PSO

(FAC-PSO) algorithm [18]. Gravitational Search algorithm (GSA) was first taken as a base by P. Li and Duan and then combined with the memory and social information concepts of PSO algorithm [19]. Comparative studies showed that proposed GSA performs better than other UAV path planners depending on default implementation of GSA or PSO algorithm [19]. Fu served a PSO algorithm that calculates the velocity of each particle by guiding the best particle of a previously determined small solution group [20].

Wang et al. developed an efficient information sharing procedure between the qualified solutions of Firefly algorithm (FA) and presented modified FA for short MFA [21]. The UAV path planning performance of MFA was investigated and a detailed comparison with other meta-heuristics such as PSO, DE, ACO, GA, a modification of GA named stud GA (SGA), Biogeography-based Optimization (BBO), Evolutionary Strategies (ES) and Population-based Incremental Learning (PBIL) was given [21]. In another study, Wang et al. proposed a three-dimensional path planning technique by hybridizing DE and Cuckoo Search (CS) algorithms [22]. The contribution of Wang et al. to the literature of meta-heuristic based path planners continued with the Bat algorithm (BA) supported by the mutation operator of DE [23]. Wang et al. also developed a BA referenced three-dimensional path planner called improved BA (IBA) and compared IBA with basic BA over the visual representations of the calculated paths [24]. For further improving the path planning performance of FA, C. Liu et al. decided to adjust the parameter related with the attractiveness of fireflies adaptively [25]. Zhu and Duan assisted the BBO algorithm with the predator–prey concept and chaos theory [26]. The novel implementation of BBO, Chaotic Predator-Prey BBO (CPPBBO), was employed to plan UAV paths by considering the constraints about the yawing angle and total flight length [26]. Black Hole (BH) algorithm was guided by Heidari and Abbaspour for UCAV path planning [27]. Glowworm Swarm Optimization (GSO) was varied by Tang and Zhou with the equations coming from PSO and PGSO was introduced [28]. Detailed performance investigations about PGSO informed that PGSO obtains better paths than other methods when the number of segmentation points is kept relatively small [28].

Yu et al. tested TLBO for the planning a UAV being operated in a fixed altitude environment and proved its competitive performance against PSO, DE, ABC and GSO algorithms [29]. The mathematical model of the path planning problem was tuned with the difficult constraints about the various enemy threats such as anti-air guns, missiles, radars, terrain and non-flight zones, turning angle, climbing or gliding slope, flight altitude and total length by Zhang and Duan and they used a DE algorithm with α level comparison based constraint-handling approach [30]. Zhou et al. designed a hybrid path planner by using Wolf Colony Search (WCS) and Complex method [31]. Duan and Qiao solved path planning problem with Pigeon-Inspired Optimization (PIO) algorithm [32]. B. Li et al. presented Balance-Evolution Strategy ABC algorithm for short BE-ABC in which trial counters are controlled when generating candidate solutions [33]. Zhang and Duan integrated the predator–prey concept into the PIO algorithm and tried to plan UCAV paths in a battlefield for which danger zones move dynamically [34]. The searching strategy of PSO algorithm using a kind of memory and mutation operator of GA that provides an extra support for avoiding solutions matched with the local optimums were referenced by Yongbo Chen et al. and a variant of the Central Force Optimization (CFO) was announced for planning of a rotary wing vertical take-off and landing (VTOL) system [35].

Zhou et al. utilized from the quantum gates for improving the performance of Wind Driven Optimization (WDO) algorithm and developed quantum WDO (QWDO) [36]. How the standard GWO performs on planning UAV paths was analyzed by Zhang et al. with two-dimensional battlefields [37]. The path planning capabilities of PSO algorithm was further improved by Liu et al. with adaptive sensitivity decision area method in which the high potential particles are determined and other candidates are removed to overcome the difficulties about the premature convergence [38]. In addition to this, Liu et al. addressed the defects of standard PSO algorithm to do with the trapping local optimums and slow convergence by Spatial Refined Voting Mechanism (SRVM) [39]. They further managed the possible collision with a newly introduced spatial-temporal collision avoidance technique when planning multiple UAVs by employing the mentioned PSO variant [39]. Luo et al. changed the representation of solutions in BA with quantum encoding and replaced the existing update and mutation models with quantum rotation gate and quantum not gate for a new path planner [40]. Q. Zhang et al. referenced Collection Decision Optimization algorithm (CDOA) and investigated its performance as a path planner [41].

Alihodzic et al. focused to solve path planning problem with Elephant Herd Optimization (EHO) algorithm [42]. In another study, Alihodzic et al. determined the number of sparks and exploitation amplitude of Fireworks (FW) algorithm when designing a UCAV path planner [43]. Miao et al. combined advantageous sides of Simplex method and SOS and provided a rich set of experimental results about their technique for battlefields with static and random enemy threats [44]. Dolicanin et al. announced a Brain Storm Optimization (BSO) algorithm based path planner [45]. Pan et al. decided to adjust the fraction probability and scaling factor of CS algorithm with the sequences of Circle-type Chaotic Map and illustrated the better path planning capabilities of the mentioned CS implementation [46]. The valuable contribution of Pan et al. is not only limited with the CS based

path planner. Pan et al. also utilized from Whale Optimization algorithm (WOA) after remodeling the encircling or searching procedures [47]. Another study successfully completed by Pan et al. was devoted to the development of CIJADE that brings together strong properties of two DE variants called CIPDE and JADE [48]. Position update procedure of BA was altered by Lin et al. with the guidance of APF [49]. They also used optimal success rate and chaos theory for further improving the local search performance of their path planner [49]. Qu et al. assumed that the members of GWO are the train agents of Reinforcement Learning (RL) and RLGWO was offered for calculating UAV paths [50]. The promising performance of GWO became source of inspiration to Qu et al. for another study in which GWO and SOS are coupled to develop a planner known as HSGWO-MSOS [51].

Yi et al. regenerated a set of low quality solutions in Monarchy Butterfly Optimization (MBO) with quantum operations and quantum inspired MBO (QMBO) was introduced [52]. Wu et al. reported an intelligent initialization schema by considering the physical limitations of the UAV being planning for ABC algorithm [53]. A Flower Pollination algorithm (FPA) guided path planner was declared by Yang Chen et al. and compared with the well-studied techniques including A*, APF and RRT [54]. After a detailed comparison between the path planning performances of BA, ABC, DE, FA, GWO, PSO, WOA, CS, a recent variant of MBO known as GSMBO, Harmony Search (HS) and Spider Monkey Optimization (SMO), Zhu et al. concluded that SMO plans more safe paths [55]. The performance of SMO tried to be further improved by Zhu et al. and Cooperation Co-evolution SMO (CESMO) was developed [56]. In order to plan a UAV path for a battlefield containing specifically designed enemy weapons and climate effects, Zhou et al. offered improved BA (IBA) [57]. Wu et al. serviced the Zaslavskii chaos map and a path planner called chaotic PSO was presented [58]. H. Xu et al. changed the critical stages of GSA by using adaptive alpha-adjusting strategy and Cauchy mutation for optimizing the interactions with enemy threats, reducing the total flight length and turning angles of a UAV [59].

Jiang et al. nearly fixed all details about the workflow of GWO by designing an efficient communication mechanism and ϵ-level comparison for handing constraints [60]. The higher number of waypoints can increase the sensitivity of the calculated paths. However, increasing the number of segmentation points or waypoints bring extra difficulty and computational burden to the path planning problem. Jarray et al. tried to handle the mentioned complexity by integrating Cooperative Co-evolution mechanism that depends on splitting the decision variables or parameters of the problem into subgroups for solving them independently into parallel GWO algorithm [61]. Du et al. tried to address some troublesome stages of Chimp Optimization algorithm (ChOA) by inspiring mathematical models of Monkey algorithm and improved ChOA for short IChOA was designed [62]. The performance of IChOA was investigated over the numerical benchmark functions in addition to the three-dimensional UAV path planning problem [62]. Wang et al. concerned with the exploration capability and convergence speed of Mayfly algorithm (MA) and developed modified MA (modMA) in which exponent decreasing inertia weight strategy, adaptive Cauchy mutation and enhanced crossover operation are combined together [63]. Some experiments using two battlefields with eight and ten enemy threats showed that modMA is not only better than MA but also more stable than PSO, GWO and Butterfly Optimization algorithm (BOA) [63]. Niu et al. replaced commonly used neighborhood topologies such as star and ring with an approach called adaptive neighborhood search for their AEO based path planner [64]. Also, they improved the decomposition stage of AEO with a dynamic method selection technique and integrated quadratic interpolation for further enhancing the search capability [64]. The new AEO algorithm referenced path planner named NSEAEO was compared with the GA and PSO variants, an improved version of TLBO (ECTLBO), GA, HSGWO-MSOS, MFO, SSA, SBO, SCA, PFA in addition to the AEO over two and three-dimensional battlefield scenarios [64]. The investigations about the capabilities of AEO based path planners were continued by Niu et al. and they designed an adaptive mechanism that controls the distance between the current optimal and newly generated candidate solution and integrated it into the workflow of AEO algorithm [65]. Five complex three dimensional scenarios were generated to evaluate the performance of recent AEO on the path planning and experimental studies proved that the modifications significantly improve the search characteristics and allow to obtain better solutions than the solutions of GA, PSO, GWO, WOA, AEO, HSGWO-MSOS, IChOA and improved adaptive GWO (AGWO) based path planners [65].

The battlefield model was tried to be simplified with a creative idea of Jia et al. when solving path planning problem by executing their special PSO algorithm [66]. The specialized PSO algorithm or DLCRPSO utilized from a rotation strategy that improves the search efficiency for high-dimensional space [66]. Search and Rescue (SAR) optimization algorithm was combined with a heuristic crossover (HC) strategy that adjusts the range for improving the efficiency of candidate generation mechanism by C. Zhang et al. and HC-SAR was introduced [67]. Comparative studies between HC-SAR and other path planners based on SAR, DE, SSA, Ant Lion Optimizer (ALO) and Squirrel Search algorithm (SSA) showed that HC-SAR can serve as a consistent UAV path planner [67]. Ait-Saadi et al. used Simulated Annealing (SA) and Singer chaotic map with Aquila Optimization

(AO) algorithm for developing Chaotic Aquila Optimization Simulated Annealing (CAOSA) and tested it on solving both two and three-dimensional UAV path planning problem [68]. Another GWO based path planner was announced by Yu et al. for which the characteristics of alpha, beta and delta wolves are changed in a manner that they search around the alpha wolf and the characteristics of the omega wolves are changed in a manner that they search around the best three wolves for enriching the exploitation [69]. Chowdhury and De introduced a GSO algorithm based path planner known as Reverse GSO (RGSO) in which the movements between the solutions represented with different glowworms are adjusted by checking the luciferin values of them [70].

Chen et al. first replaced the Cartesian coordinate system with a spherical coordinate system in which some constraints about the angle and velocity of a UAV are handled more clearly when planning a flight path [71]. They also introduced a mechanism called Truncated Mean Stabilization (TMS) for maintaining the population diversity by replacing some solutions with a newly determined one. The modernized BA that uses spherical coordinate system and TMS approach was named TMS-SBA by Chen et al. and tested for calculating paths in four different battlefield scenarios [71]. The potential of changing the Cartesian coordinate system to realize the characteristics of a UAV or UCAV was also utilized by Huang et al. for designing a new PSO algorithm based path planner, Adaptive Cylinder Vector PSO with DE (ACVDEPSO), and ACVDEPSO was experimented in three-dimensional environments generated with the Digital Elevation Model (DEM) maps [72]. Hu et al. improved the overall optimization performance of the standard Honey Badger algorithm (HBA) by invoking Bernoulli shift map for initialization of the population, piecewise optimal decreasing neighborhood for stabilizing the unbalanced convergence characteristics and finally horizontal crossing with strategy adaptation for the generation of new candidates [73]. They tested new HBA variant to plan UAV paths in different battlefield environments containing only circular or irregular obstacles [73].

2 Mathematical model of path planning problem

The calculation of a path for a UAV, UCAV or other similar aerial vehicle requires a strong mathematical description about the different enemy threats, their sensing, detecting or shooting capabilities, fuel consumption or battery usage and finally kinematic constraints on the turning and climbing maneuvers. In addition to the mathematical descriptions of the enemy threats, fuel or battery usage and kinematic constraints, a model that defines how a path can be generated and a score calculation schema for deciding which path is more better should also be supplied. Given that a UAV or UCAV starts flight from the point $P_s = (x_s, y_s, z_s)$ to find or destroy a target located at the point $P_t = (x_t, y_t, z_t)$ and a reference line between the P_s and P_t is drawn by considering the xy-plane.

When the operations to do with the drawing of a reference line are completed, it is divided equally into $D + 1$ segments by using D segmentation points [64]. Each segmentation point on the reference line is actually responsible for intersecting with a unique line that is perpendicular to the reference line. If the lines, each is perpendicular to the reference line and intersects only one segmentation point, are organized, a set of lines showed as $L = \{L_1, L_2, \ldots, L_{D-1}, L_D\}$ can be obtained. The set L in which L_1 corresponds to the vertical line passing through the first segmentation point, L_2 corresponds to the vertical line passing through the second segmentation point, and so on opens a gate for the subsequent operations of the path planning. If only one point on each line in the set L is selected and then combined with the P_s and P_t by guiding that the P_s is the start point and P_t is the target point, a set of points or $P = \{P_s, P_1, \ldots, P_D, P_t\}$ and an implicit path after connecting sequential pair of points in set P with a line segment are generated [64].

The method lying behind the definition of a UAV or UCAV path through the set L and set P depends on strong mathematical and geometrical backgrounds. All the lines in the set L require correct equations that satisfy the prerequisites about the segmentation points and reference line between P_s and P_t. Also, it must be guaranteed that each point of the set P is selected in a manner that the point P_i is on the line L_i where i ranges from 1 to D and huge amount of computational burden arises. In order to reduce the computational effort about the sets of lines and points, an appropriate coordinate system transformation that converts the reference line into the horizontal axis of the new coordinate system by referencing Eq. (1) can be used [64]. In Eq. (1), x_k, y_k and z_k represent the x-axis, y-axis and z-axis values of point P_k located at the original coordinate system, while $\acute{x}_k$, $\acute{y}_k$ and $\acute{z}_k$ represent the $\acute{x}$-axis, $\acute{y}$-axis and $\acute{z}$-axis values of point $\acute{P}_k$ and $\acute{P}_k$ corresponds the transformed counterpart of point P_k for the new coordinate system. Finally, θ is matched with the angle of rotation and calculated as $arctan((y_k - y_s)/(x_k - x_s))$.

$$\begin{bmatrix} \acute{x}_k \\ \acute{y}_k \\ \acute{z}_k \end{bmatrix} = \begin{bmatrix} cos(\theta) & sin(\theta) & 0 \\ -sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_k - x_s \\ y_k - y_s \\ z_k \end{bmatrix} \quad (1)$$

One of the first advantages coming with the mentioned coordinate transformation is about the $\acute{x}$-axis values of the corresponding points in the set P. Because of each line in the set L is vertical to the reference line or horizontal axis of the

new coordinate system and the distance between the subsequent pair of lines is equal, $\acute{x}$-axis value of any point on the line $\acute{L}_i$ where $\acute{L}_i$ shows the counterpart of L_i in the set L for new coordinate system can be calculated as $i|P_sP_t\ |/(D+1)$. If the $\acute{y}$-axis values of the points on the lines in set L are selected and they are brought together with the vertical axis values of the transformed start and target points such as $\{\acute{y}_s, \acute{y}_1, \acute{y}_2, \ldots, \acute{y}_{D-1}, \acute{y}_D, \acute{y}_t\}$, path planning can be turning into a D-dimensional optimization problem that requires minimization of difficult objectives about enemy threats, battery or fuel consumption measured over total flight length, turning and climbing angles. In Fig. 1, how the set of lines and set of points are utilized to represent a path and transitions between the initial and original coordinate systems are carried out is illustrated over a battlefield with four enemy threats.

A relatively small modification on one of the guessed points in the set P can cause a dramatic change for the corresponding path and a quality or score calculation schema taking into account the enemy threats and their properties, fuel or battery consumption, turning and climbing angles

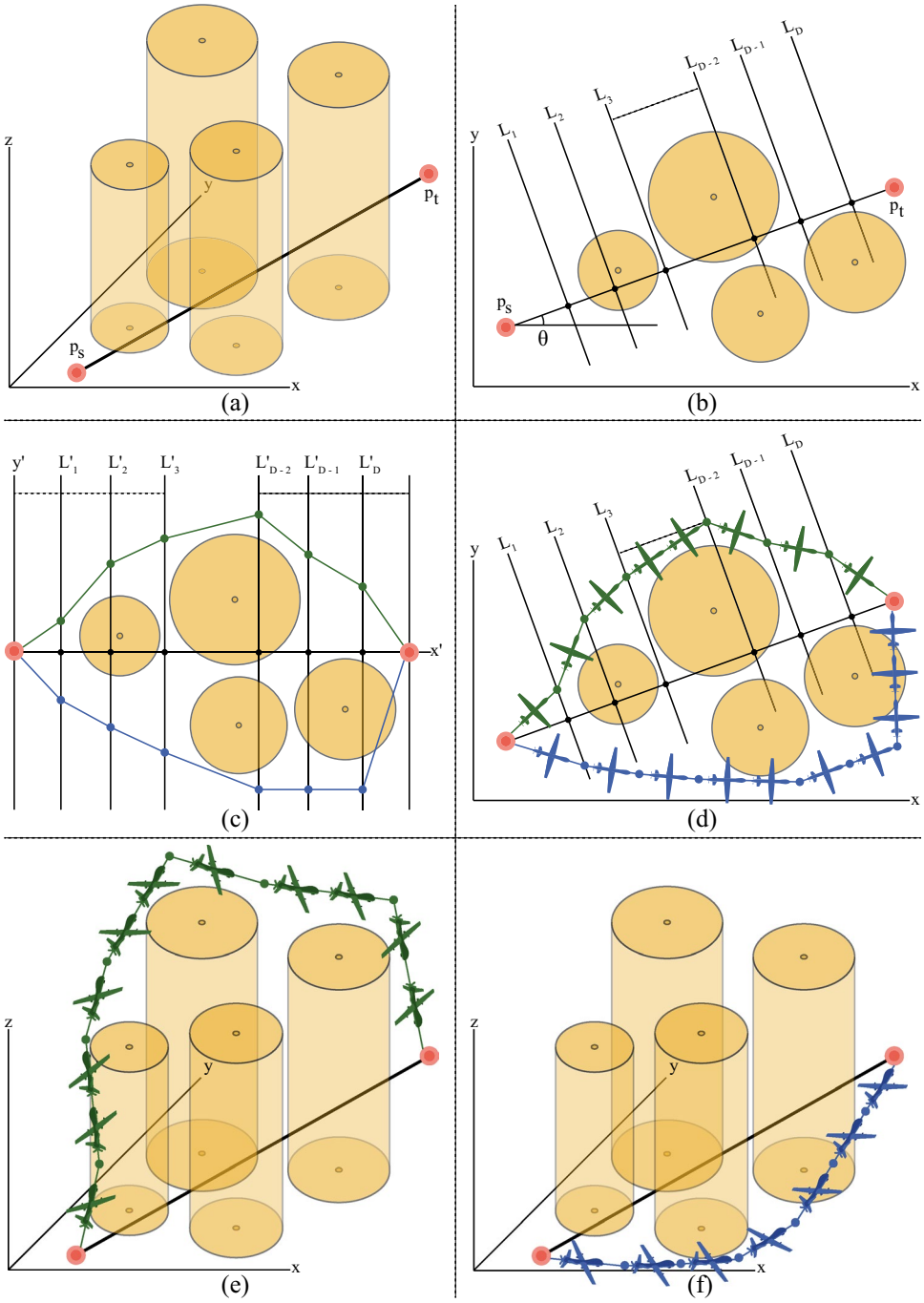

Fig. 1 A three-dimensional battlefield (**a**), *xy*-plane (**b**), transformed counterpart (**c**), determined paths (d) and their provisions to the original battlefield (**e**)–(**f**)

should be used to understand the appropriateness of a path and make a discrimination between more than one candidates as described in Eq. (2) [64]. In Eq. (2), C_t is used on behalf of the cost of all enemy threats and it is calculated by taking the integral of w_t from 0 to ℓ where ℓ shows the total length of the discovered path. Similarly, C_f is used on behalf of the cost of fuel or battery consumption of UAV or UCAV system and calculated by taking the integral of w_f from 0 to ℓ. While C_s represents the cost of kinematic limitations of considered aerial vehicle, it has two different parts one of which is related with the turning angles and the other is related with the climbing angles. Also, it should be noticed that C_t, C_f and C_s are weighted with λ_t, λ_f and λ_s whose sum is equal to 1 for adjusting their contributions on the total path cost showed as C.

$$C = \lambda_t C_t + \lambda_f C_f + \lambda_s C_s = \lambda_t \int_0^{\ell} w_t d\ell + \lambda_f \int_0^{\ell} w_f d\ell + \lambda_s \left(\sum_{j=1}^{D} \varnothing_j + \sum_{j=1}^{D+1} \Psi_j \right) \tag{2}$$

The integral calculations to do with the C_f can be simplified by executing an accurate approximation. Because of the fuel consumption or battery usage of a UAV or UCAV is directly proportional to the length of path, w_f can be replaced with a constant such as 1 [64]. An accurate but more detailed approximation can also help the integral calculations about C_t. Given that P_i and P_j are two adjacent points in the set P and the length of line segment between these points is found as L_{ij}. Also, it is noted that the line segment of length L_{ij} is divided into ten equal subsegments with the help of nine subsegmentation points and then the first, third, fifth, seventh and ninth subsegmentation points are selected and called as 0.1, 0.3, 0.5, 0.7, 0.9 subsegmentation points. If the line segment of length L_{ij} is in the effect range of *kth* enemy threat with the grade t_k, the cost of considered enemy threat for the line segment between P_i and P_j or $C_{t,(ij),k}$ is found by using Eq. (3) [64]. While the Euclidean distance between 0.1 subsegmentation point and the center of the *kth* enemy threat is represented with $d^4_{0.1,i,k}$ in Eq. (3), the Euclidean distances between the other selected segmentation points and the center of the *kth* enemy threat are showed with $d^4_{0.3,i,k}$, $d^4_{0.5,i,k}$, $d^4_{0.7,i,k}$ and $d^4_{0.9,i,k}$ for the same equation. After calculating the cost of each enemy threat for all of the line segments and then summing them, C_t is approximated successfully.

$$C_{t,(ij),k} = \frac{L_{ij} t_k}{5} \left(\frac{1}{d^4_{0.1,i,k}} + \frac{1}{d^4_{0.3,i,k}} + \frac{1}{d^4_{0.5,i,k}} + \frac{1}{d^4_{0.7,i,k}} + \frac{1}{d^4_{0.9,i,k}} \right) \tag{3}$$

For tracking the calculated path, a UAV or UCAV should perform different maneuvers that necessitate aggressive turning and climbing with variable angles. However, a UAV or UCAV system has certain limitation about the turning and climbing angles and they should be considered when the overall path quality is calculated. Assume that three subsequent points such as P_j, P_{j+1} and P_{j+2} are selected from the set P and two vectors such as $\overrightarrow{p_j p_{j+1}}$ and $\overrightarrow{p_{j+1} p_{j+2}}$ are generated by referencing these points. When P_j, P_{j+1} and P_{j+2} are the first three points of set P, they correspond to P_s, P_1 and P_2. In a similar manner, when P_j, P_{j+1} and P_{j+2} are the last three points of set P, they correspond to P_{D-1}, P_D and P_t. The calculation of turning angle or $\varnothing_j$ by considering the P_j, P_{j+1} and P_{j+2} points and $\overrightarrow{p_j p_{j+1}}$ and $\overrightarrow{p_{j+1} p_{j+2}}$ vectors can be made with Eq. (4). If the absolute value of the $\varnothing_j$ is less than or equal to the maximum angle or $\varnothing_{max}$ of the UAV being operated, the effect of turning with the angle of $\varnothing_j$ is simply ignored for the C_s. Otherwise, absolute value of the calculated turning angle is summed with the absolute values of other turning angles violating the constraint about the maximum turning angle.

$$\varnothing_j = arctan \left(\frac{\| \overrightarrow{p_j p_{j+1}} \times \overrightarrow{p_{j+1} p_{j+2}} \|}{\overrightarrow{p_j p_{j+1}} \cdot \overrightarrow{p_{j+1} p_{j+2}}} \right) \tag{4}$$

For calculating the climbing angle, subsequent points taken from set P and some vectors are needed. Assume that two subsequent points namely P_j and P_{j+1} are selected from set P and $\overrightarrow{p_j p_{j+1}}$ is the vector generated by referencing these points. When P_j and P_{j+1} are the first two points of set P, they correspond to P_s and P_1. In a similar manner, when P_j and P_{j+1} are the last two points of set P, they correspond to P_D and P_t respectively. The calculation of climbing angle or Ψ_j by considering P_j, P_{j+1} points and $\overrightarrow{p_j p_{j+1}}$ vector can be made with Eq. (5). If the absolute value of $\Psi_j - \Psi_{j-1}$ operation is less than or equal to the maximum climbing angle or Ψ_{max} of the UAV, the effect of climbing is simply ignored for the C_s. Otherwise, the absolute value of $\Psi_j - \Psi_{j-1}$ operation is summed with the absolute values of other climbing angle calculations violating the constraint about the maximum climbing angle. The whole symbols used for the description and formulation of the path planning problem can be accessed in Table 1.

$$\Psi_j = \arctan \left(\frac{Z_{j+1} - Z_j}{\| \overrightarrow{p_j p_{j+1}} \|} \right) \tag{5}$$

Table 1 Used symbols and their descriptions for path planning

Symbols	Description
D	Number of segmentation points or parameters
P_s, P_t	Start and target points
L, L_i	Line set and its *ith* member
P, P_i	Point set and its *ith* member
θ	Rotation angle for coordinate transformation
$\acute{L}_i$	Counterpart of L_i for new coordinate system
C	Total cost of path
C_f	Cost of fuel consumption
C_t	Cost of enemy threats
C_s	Cost of turning and climbing maneuvers
$\lambda_f, \lambda_t, \lambda_s$	Weighting factors for C_f, C_t and C_s
ℓ	Total length of path
L_{ij}	Length of line segment between P_i and P_j points
$\varnothing_j$	Cost of turning for P_j, P_{j+1} and P_{j+2} points
Ψ_j	Cost of climbing for P_j and P_{j+1} points
$\varnothing_{\max}, \Psi_{\max}$	Maximum turning and climbing angles

3 Immune plasma algorithm

The immune system tries to protect a host by increasing the amount of plasma cells and their synthesis products also called antibodies. Antibodies are actually a type of proteins and can circulate in the blood as free-floating forms [74, 75]. When an antibody detects an antigen for which the antibodies are produced specially, it binds that antigen with the purpose of inactivating antigen functionalities. However, some persons who suffering from the immune system disorders have several difficulties for synthesizing remarkable amount of antibodies and when they are infected, hospitalization and intense care can be needed [74, 75]. In order to help the treatment operations of a critical person, the antibody rich part of the blood donated by an individual recovered previously can be utilized successfully.

The immune or convalescent plasma treatment is one of the strong medical methods guiding the fact that the antibodies can be transferred from the recovered individual or individuals to the critical patients or receivers and its efficiency was proven against the great influenza of 1918 pandemic more than a century ago and the recent global COVID-19 crisis [75, 76]. When the details of the immune or convalescent plasma treatment is controlled carefully, it is seen that there is an obvious analogy with the main operations known as exploration and exploitation of a meta-heuristic algorithm. By considering the mentioned analogy, Aslan introduced a new intelligent optimization technique called IP algorithm or for short IPA [7]. In IP algorithm, each person or individual of the population represents a possible solution of the optimization problem being solved. An infection can spread easily among the members of population and their immune responses are calculated according to the objective or cost function of the problem. While an individual with small objective function value corresponds to a qualified solution for a minimization problem, an individual with high objective function value corresponds to a qualified solution for a maximization problem [7]. Some individuals representing poor solutions are labeled as receivers and tried to be treated with the plasma taken from other individuals that are selected as donors because of their high quality immune responses. The mathematical models used by IP algorithm for distributing infection in the population, selecting receiver and donor individuals, applying plasma treatment and controlling the immune memories of donors were stated in the following subsections.

3.1 Initializing the members of population

Population based meta-heuristics such as IP algorithm starts the search operations by generating a set of solutions randomly. Given that IP algorithm with the population of size PS is employed for solving a D-dimensional optimization problem, *kth* individual also termed as x_k can be initialized by using Eq. (6) [7]. In Eq. (6), x_{kj} is matched with the jth parameter for which the lower and upper bounds are $x_j^{\min}$ and $x_j^{\max}$. Also, it should be noticed that $rand(0, 1)$ is a random number taking its value between 0 and 1.

$$x_{ij} = x_j^{\min} + \text{rand}(0, 1)(x_j^{\max} - x_j^{\min}) \tag{6}$$

3.2 Infecting the members of population

In an infection cycle of IP algorithm, there is a stage that is responsible for distributing infection from one individual to another with Eq. (7) where x_k is the individual being infected by the randomly selected x_m individual [7]. Moreover, it should be noted that x_{kj} and x_{mj} are the jth parameters of them and the j index is determined randomly from the set $\{1, 2, \ldots, D\}$. For representing the infectious x_k individual, a temporary solution or x_k^{inf} is used in the same equation. All of the parameters belonging to x_k^{inf} are equal to the corresponding parameters of x_k except the *jth* one and the newly calculated *jth* parameter of x_k^{inf} is symbolized with x_{kj}^{inf}.

$$x_{kj}^{inf} = x_{kj} + rand(-1, 1)(x_{kj} - x_{mj}) \tag{7}$$

The infection triggers the immune system of x_k individual and a special response in terms of antibodies is given. In order to evaluate the immune response of the infectious x_k individual or amount of synthesized antibodies, the value of the objective function f is utilized. If the immune response of the infectious x_k individual or $f(x_k^{inf})$ is less than the antibody amount of the same individual before the infection or $f(x_k)$ by considering a minimization problem, it is decided that x_k

individual is capable of handling infection and its immune memory is re-organized for a quick response to the similar infection as in Eq. (8) [7]. Otherwise, x_k individual and its *jth* parameter are left unchanged.

$$x_{kj} = \begin{Bmatrix} x_{kj}^{inf}, & if\ f(x_k^{inf}) < f(x_k) \\ x_{kj}, & otherwise \end{Bmatrix} \tag{8}$$

3.3 Applying plasma treatment

The second stage of an infection cycle in IPA is related with the selection of receivers and donors and then the application of plasma treatment. IP algorithm decides how many individuals will be receiver and how many individuals will be donors by introducing two control parameters called number of receivers or *NoR* and number of donors or *NoD* [7]. When IPA reaches the second stage of an infection cycle, it first sorts the individuals of the population by considering their objective function values in ascending order and then labels the last *NoR* individuals as critical patients or receivers and selects the first *NoD* individuals as plasma donors [7]. After determining the receiver and donor individuals, IPA starts plasma treatment. Given that x_k^{rcv} is the *kth* receiver from the receiver set of size *NoR* and x_m^{dnr} is the randomly selected donor from the donor set of size *NoD*. For the transfer of a dose of plasma from the x_m^{dnr} to the x_k^{rcv}, a mathematical model as detailed in Eq. (9) where *j* is selected sequentially from the set $\{1, 2, \ldots, D\}$ is used [7]. In Eq. (9), x_k^{rcv-p} is matched with the plasma transferred counterpart of x_k^{rcv} and *jth* parameters of them are x_{kj}^{rcv} and x_{kj}^{rcv-p}. If the $f(x_k^{rcv-p})$ is better than the $f(x_m^{dnr})$ and proves the efficiency of treatment, x_k^{rcv} is updated with the parameters of x_k^{rcv-p} and second dose of plasma is prepared. Otherwise, x_k^{rcv} is updated with the parameters of x_m^{dnr} and treatment is completed for x_k^{rcv} [7].

$$x_{kj}^{rcv-p} = x_{kj}^{rcv} + rand(-1, 1)(x_{kj}^{rcv} - x_{mj}^{dnr}) \tag{9}$$

The second or subsequent dose of plasma is transferred to x_k^{rcv} by using the mathematical model introduced for the transfer of first dose. However, in order to decide that whether the treatment will be continued with the third or subsequent dose of plasma or not, a comparison between the objective function values of x_k^{rcv-p} and x_k^{rcv} is carried out [7]. If the objective function value of x_k^{rcv} immediately after the second dose of plasma or $f(x_k^{rcv-p})$ is better than the objective function value of x_k^{rcv} before the second dose of plasma or $f(x_k^{rcv})$, x_k^{rcv} is updated with the parameters of x_k^{rcv-p} and third dose of plasma is prepared. Otherwise, the treatment of x_k^{rcv} is completed and the next receiver is selected if exists for starting the plasma transfer operations.

3.4 Updating immune memories of donors

The immune response or amount of synthesized antibodies by an individual who recovers shortly before and helps critical individuals for the treatment can change as time goes by or with the frequency of encountering to the same of similar infection. If the frequency of encountering to the infection increases with time, the immune memory recognizes the intruder quickly and a strong response in terms of synthesized antibodies is given. For integrating this type of mechanism into the workflow of the IPA, the ratio between t_{cr} and t_{max} and a random number generated between 0 and 1 were utilized [7]. While t_{cr} shows the current evaluation number and it is incremented by one for each request to the procedure calculating the objective function value, t_{max} demonstrates the maximum evaluation number and IPA terminates when t_{cr} becomes equal to t_{max}. If the ratio between t_{cr} and t_{max} is less than the generated random number, it is decided that the immune memory of the *mth* donor individual or x_m^{dnr} still continues to learn details about the intruder causing infection and an entire re-initialization as in Eq. (6) is applied [7]. Otherwise, the immune memory of the x_m^{dnr} is changed slightly by using Eq. (10) where *j* index ranges from 1 to *D* [7]. As easily seen from the decision mechanism about how the donor individual is updated, the probability of execution Eq. (10) gets higher while the IPA reaches termination and allows a donor for protecting its memory partially.

$$x_{mj}^{dnr} = x_{mj}^{dnr} + rand(-1, 1)x_{mj}^{dnr} \tag{10}$$

4 Details of hospitalization mechanism for immune plasma algorithm

As stated previously, the standard implementation of the IP algorithm completes the treatment of an x_k^{rcv} individual if the first dose of plasma does not improve the antibody response of x_k^{rcv} as better as the antibody response of x_m^{dnr} donor. Moreover, when the IP algorithm decides that the treatment of x_k^{rcv} is ended immediately after the first dose of plasma from the x_m^{dnr} donor individual, the x_k^{rcv} is updated with the corresponding parameters of x_m^{dnr} for guaranteeing that at least one dose of plasma is transferred. Even though the idea lying behind the existing treatment schema of IPA ensures that the quality of the solution represented by a receiver individual becomes equal or better than the quality of the solution represented by the selected donor, it requires subtle configuration of the *NoR* and *NoD* parameters in order to maintain the population diversity while increasing the qualities of the existing solutions.

Algorithm 1 Distribution of infection by considering hospitalization

```
x^best ← the best solution found so far
hospitalization[1...PS] ←a vector showing hospitalization status of individuals
/*Determining the number of hospitalized individuals*/
numOfHosp ← 0
for k ← 1...PS do
    if hospitalization[k] == 1 then
        numOfHosp ← numberOfHosp + 1
    end if
end for
if numOfHosp ≠ (PS − 1) then
    /*Infection distribution*/
    for k ← 1...PS do
        if hospitalization[k] == 0 then
            if t_cr < t_max then
                t_cr ← t_cr + 1 and
                x_m ←a random individual from the set of non-hospitalized individuals
                x_k^inf ← infect x_k with x_m by using Eq. (7)
                if f(x_k^inf) < f(x_k) then
                    Update x_k with x_k^inf
                    if f(x_k) < f(x^best) then
                        Update x^best with x_k
                    end if
                end if
            end if
        end if
    end for
end if
```

For further improving the performance of IPA and removing the necessity of both requirement and configuration of *NoR* and *NoD* parameters, a more efficient and realistic model by considering that a receiver or receivers can stay in a hospital rather than simply discharging them if the first dose of plasma does not generate the expected effect can be designed. At the end of the stage related with the distribution of infection in each cycle, the worst solution of the non-hospitalized individuals is first assumed as the patient and added to the set of hospitalized solutions or receivers. All of the hospitalized individuals are treated by transferring plasma. If the transferred dose of plasma gives a tremendous contribution to the antibody amount of a receiver individual, it is discharged from the hospital and becomes ready for the interaction of other healthy individuals in the next cycle. Otherwise, the mentioned receiver is isolated from other healthy individuals and stays at the hospital for the treatment operations of the subsequent cycle. When the number of hospitalized individuals or receivers is equal to $PS-1$ for a population of size PS, it is easily understood that there is only one healthy individual and the operations to do with the distribution of infection are skipped. On the other hand, if the number of hospitalized individuals or receivers is not equal to $PS-1$, non-hospitalized individuals still interact with each other and distribution of infection between these individuals can continue. In order to understand that how the discrimination between the hospitalized and non-hospitalized individuals is carried out when distributing the infection, Alg. (1) given can be examined.

Because of the newly introduced hospitalization mechanism adjusts the number of receivers dynamically for each cycle, an improved plasma treatment schema that is able to successfully handle the varying composition and number of receivers should be designed. Assume that x^{best} is the best solution found so far and x^{dnr} is the most qualified solution in the current population. For obtaining plasma being used for the treatment of hospitalized individuals, the mathematical representation given in Eq. (11) can be employed. While x_j^{pls-t} represents the newly determined jth parameter of the x^{pls} and x^{pls} corresponds to the plasma initialized with x^{best}, x_j^{dnr} shows the jth parameter of the x^{dnr} individual. If the newly calculated jth parameter or x_j^{pls-t} of the x^{pls} improves the overall quality of the collected plasma, a greedy selection between x_j^{pls-t} and x_j^{pls} is executed. After controlling all of D different parameters sequentially and applying greedy selection between the new and existing ones, x^{pls} that is at least equal to or better than the x^{best} is obtained and ready to the usage for the treatment operations.

$$x_j^{pls-t} = x_j^{pls} + rand(-1,1)(x_j^{dnr}) \tag{11}$$

Algorithm 2 Treatment of the hospitalized individuals

```
/*Collecting plasma by using new approach*/
x^dnr ← the donor from the population
x^pls ← x^best /*initialization of the x^pls */
for j ← 1 ... D do
    if t_cr < t_max then
        t_cr ← t_cr + 1
        x_j^{pls-t} ← find the new value of x_j^pls by using Eq. (11)
        if f(x^pls) ≥ f(x^best) then
            Convert the x_j^pls into its previous value
        else
            Update x^best with x^pls
        end if
    end if
end for
/*Determining the number of hospitalized individuals*/
numOfHosp ← 0
for k ← 1 ... PS do
    if hospitalization[k] == 1 then
        numOfHosp ← numOfHosp + 1
    end if
end for
if numOfHosp < (PS − 1) then
    numOfHosp ← numOfHosp + 1
    index ←index of the worst individual of the non-hospitalized ones
    hospitalization[index] ← 1 /*hospitalization of the worst solution*/
end if
for k ← 1 ... PS do
    if hospitalization[k] == 1 then
        x_k^rcv ← get the receiver individual
        j_rand← a random integer between 1 and D
        if t_cr < t_max then
            t_cr ← t_cr + 1
            for j ← 1 ... D do
                if j ≠ j_rand then
                    Set x_kj^{rcv−p} by using x_j^pls
                else
                    x_kj^{rcv−p} ← calculate the new value of x_kj^rcv by using Eq. (12)
                end if
            end for
            if f(x^{rcv−p}) < f(x^pls) then
                Update x_k^rcv with x_k^{rcv−p}
                hospitalizaton[k] ← 0 /*discharging x_k^rcv from the hospital*/
            end if
            if f(x_k^{rcv−p}) < f(x_best) then
                Update x_best with x_k^{rcv−p}
            end if
        end if
    end if
end for
```

The default workflow of the IP algorithm uses the selected donor as the source of plasma and the treatment is modeled in a manner that each parameter of a receiver is changed with the information provided by its donor. However, in order to utilize from the information provided by the donor or collected plasma, some parameters of the receiver should be set to the corresponding parameters of the donor or collected plasma directly while the remaining ones are changed appropriately. A more efficient transfer approach that focuses on increasing the positive contribution of the treatment can be formulated as in Eq. (12). In Eq. (12), j_{rand} is used on behalf of a random number generated between 1 and D and it is compared with the j index chosen sequentially from the set $\{1, 2, \ldots, D\}$. If the j_{rand} is found equal to the current value of j index, jth parameter of the x_k^{rcv-p} or x_{kj}^{rcv-p} is calculated again with the help of corresponding parameter of x^{pls}. Otherwise, x_{kj}^{rcv-p} is set to the jth parameter of the x^{pls} or x_j^{pls} for a direct utilization from the valuable information provided by the x^{pls}. When the transfer of plasma to the x_k^{rcv} is completed and the antibody level of x_k^{rcv} immediately after the treatment calculated as $f(x_k^{rcv-p})$ is determined, a simple comparison between $f(x_k^{rcv-p})$ and the antibody level of x^{pls} also calculated as $f(x^{pls})$ is carried out. If $f(x_k^{rcv-p})$ is better than $f(x^{pls})$, x_k^{rcv} is updated with the x_k^{rcv-p} and x_k^{rcv} is discharged from the hospital. Otherwise, x_k^{rcv} continues to stay at hospital and waits the treatment operations of the next cycle.

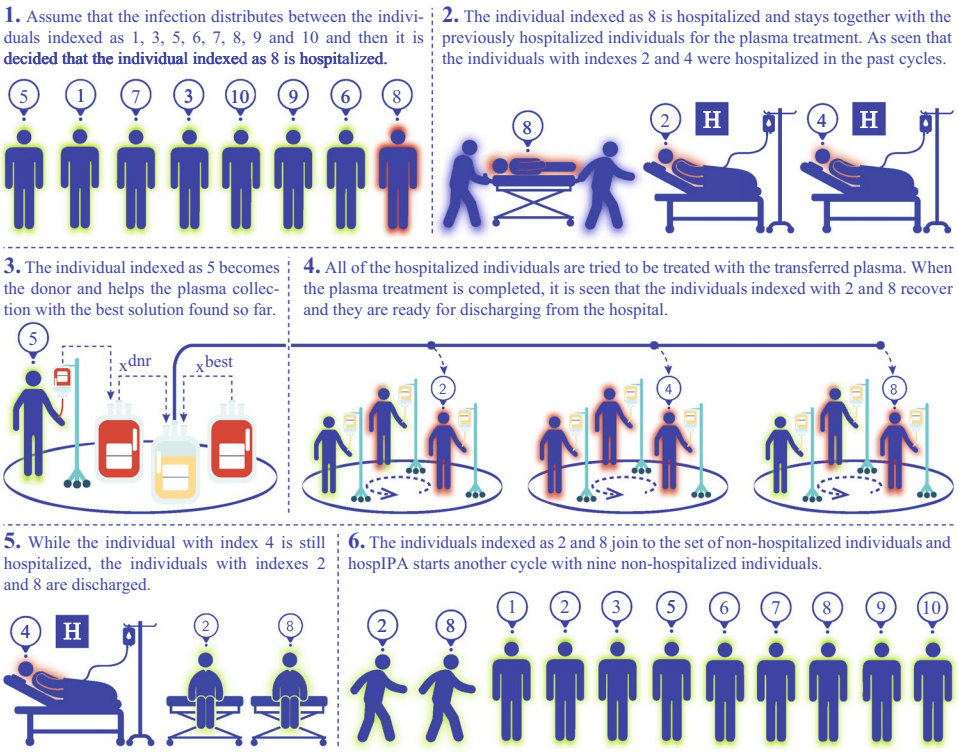

Fig. 2 A pictorial description of fundamental operations in hospIPA

The details of the proposed treatment schema for the hospitalized individuals are presented in Alg. (2).

$$x_j^{rcv-p} = \begin{cases} x_j^{pls}, & if\ j \neq j_{rand} \\ x_j^{rcv} + rand(-1,1)(x_j^{rcv} - x_j^{pls}), & otherwise \end{cases} \quad (12)$$

The IP algorithm that integrates the hospitalization mechanism into the workflow of it to dynamically determine the number of individuals who will be treated as receivers and the specialized plasma generation and transfer schema is named hospital IPA for short hospIPA. In the hospIPA, there is no need to the *NoR* and *NoD* parameters and their subtle adjustments. Because of the hospitalized individuals are not allowed to interact with the non-hospitalized individuals, the vicinity of the qualified solutions represented by the non-hospitalized individuals is explored more successfully. Also, it should be noticed that the treatment schema of a hospitalized or receiver individual is re-designed completely for handling the difficulty of dynamically determined set of receivers and increasing the effectiveness of plasma collection and transfer operations. When the operations related with the collection of plasma and its transfer to the receiver or receivers are carried out, hospIPA gets a chance of exploiting the solutions corresponding to the x^{best}, x^{dnr} and x^{pls} implicitly. In Fig. 2, a hypothetical scenario with ten individuals was illustrated to describe the hospitalization mechanism and treatment schema of the hospIPA.

5 Experimental studies

The quality of a path being calculated by hospIPA changes according to the values of the algorithm specific control parameters, properties of the battlefields, enemy threats and finally number of segmentation points. Moreover, extra mechanisms executed by hospIPA effect the execution time

Table 2 Details of battlefields used for fixed altitude path planning

Sc	Threat centers	Threat radius	Threat grade	Start-Target point
1	(12,48),(24,33),(27,58),(30,70), (55,80),(59,52),(70,34),(70,65)	12,9,9,10, 9,10,12,7	1,12,3,2, 7,9,13,5	(10,15) (80,75)
2	(20,70),(25,19),(25,39),(45,20), (47,41),(50,61),(70,53),(75,74),(78,20)	20,9,9,9, 9,9,9,9,20	7,5,5,5, 5,5,5,5,7	(5,5) (95,95)
3	(10,50),(20,20),(30,42),(30,80),(50,55), (60,10),(60,80),(65,38),(75,65),(90,80)	10,9,8,10,10, 10,10,11,8,10	8,6,5,4,7, 6,7,6,8,10	(10,0) (80,100)

Table 3 Results of hospIPA with varying *PS* values for Scenario-1

D		*PS*				
		30	40	50	75	100
10	Best	38.394	38.378	38.366	38.367	38.392
	Worst	50.915	38.443	38.403	38.435	38.431
	Mean	39.657	38.403	**38.386**	38.411	38.409
	Std	3.817	0.023	0.016	0.025	0.017
15	Best	38.256	38.262	38.283	38.278	38.306
	Worst	38.335	55.448	62.381	55.472	50.426
	Mean	**38.285**	41.709	43.107	41.744	40.362
	Std	0.028	6.987	9.802	6.981	4.578
20	Best	38.245	38.259	38.296	38.443	38.359
	Worst	38.350	56.749	47.276	56.419	59.664
	Mean	**38.290**	39.563	39.250	49.477	44.374
	Std	0.038	4.672	2.722	7.153	6.921
25	Best	38.363	38.401	38.472	38.533	49.943
	Worst	46.183	63.416	77.490	82.181	63.363
	Mean	**41.732**	45.365	61.429	58.621	55.508
	Std	3.852	8.956	12.009	16.920	5.172

Bold values show the better results

Table 4 Results of hospIPA with varying *PS* values for Scenario-2

D		*PS*				
		30	40	50	75	100
10	Best	57.584	57.785	57.559	57.644	57.698
	Worst	63.757	61.398	63.962	60.526	60.298
	Mean	60.632	59.123	60.316	**58.356**	59.276
	Std	2.580	1.728	2.544	1.219	1.264
15	Best	54.412	54.215	54.195	54.211	56.364
	Worst	58.646	58.367	58.573	56.448	60.634
	Mean	56.493	56.200	56.204	**55.365**	58.323
	Std	1.132	1.541	1.503	1.064	1.374
20	Best	53.813	53.756	53.677	53.998	53.767
	Worst	64.771	57.127	64.403	64.728	57.217
	Mean	55.374	54.944	56.904	56.253	**54.304**
	Std	3.751	1.170	4.950	3.083	1.164
25	Best	53.648	53.744	53.595	53.721	53.686
	Worst	63.181	63.156	60.563	64.823	62.755
	Mean	55.984	56.352	**55.405**	55.721	57.713
	Std	2.722	2.611	2.209	2.816	2.200

Bold values show the better results

and convergence performance of the same algorithm. For a more organized investigations about the path planning performance of hospIPA, the whole experimental studies were divided into four subsections. While the first and second subsections were devoted to the tests and comparative studies for the two and three-dimensional battlefield scenarios, some results about the execution times of hospIPA were shared in the third subsection. Finally, the convergence characteristics of hospIPA and statistical significance of its solutions were evaluated in the fourth subsection.

Table 5 Results of hospIPA with varying *PS* values for Scenario-3

D		*PS*				
		30	40	50	75	100
10	Best	49.772	49.772	49.772	49.771	49.779
	Worst	49.797	49.785	49.796	49.786	49.789
	Mean	**49.776**	**49.776**	49.781	**49.776**	49.785
	Std	0.009	0.004	0.010	0.004	0.004
15	Best	49.763	49.756	49.750	49.763	49.769
	Worst	49.815	49.851	49.957	49.872	49.850
	Mean	**49.779**	49.785	49.785	49.815	49.801
	Std	0.016	0.034	0.062	0.042	0.029
20	Best	49.789	49.786	49.825	49.867	49.945
	Worst	49.927	50.075	50.383	50.213	52.334
	Mean	**49.845**	49.915	49.964	50.011	50.732
	Std	0.055	0.113	0.176	0.088	0.900
25	Best	49.878	49.932	50.103	49.869	50.045
	Worst	62.445	55.805	53.399	57.162	52.150
	Mean	50.831	**50.611**	52.194	51.529	50.860
	Std	3.158	1.443	1.078	2.151	0.689

Bold values show the better results

5.1 Planning paths for two-dimensional battlefields with hospIPA

The path planning performance of hospIPA after assuming that the altitude is fixed was investigated over three different battlefield scenarios each has four test cases generated by setting the number of segmentation points or *D* as 10, 15, 20 and 25. An enemy threat in a battlefield used for experiments is represented with a circle and the location of the circle center and radius are decided previously. Moreover, grades were assigned to the enemy threats for defining danger levels of them. The details about the battlefields and included enemy threats were given in Table 2 [64]. Because of only the population size or *PS* is adjustable for hospIPA, each test case was experimented by setting *PS* to 30, 40, 50, 75 and 100. A run of hospIPA that is terminated when the evaluation counter reaches to 6000 was repeated 30 times with random seeds and the best solution and its objective function value found at the end of a run were recorded [64]. By using the recorded objective function values, the best, worst, mean best objective function values and standard deviations were determined and then summarized in Table 3 for Scenario-1, Table 4 for Scenario-2 and finally Table 5 for Scenario-3.

The results given in Tables 3, 4, 5 provide important information about the relatively stable and consistent performance of hospIPA. When the value being assigned to *PS* parameter is increased, a population based meta-heuristic can discover the search space more efficiently at the initial stage of a run and start subsequent operations with a set of solutions providing required diversities. However, the number of function evaluations spent per cycle, iteration or generation is directly proportional to the population size and a meta-heuristic terminates more quickly when its population size is set to higher values without executing algorithm specific search processes. On the other hand, when the population is configured with a small set of solutions, algorithm continues to search more longer and the probability of finding qualified solutions is boosted intrinsically. Even though the small set of solutions brings some advantages to the considered algorithm by allowing it for showing exclusive exploration and exploitation characteristics, the diversity of solutions can not be enough to represent the different regions of the space and convergence problems can arise from one run to another. As stated earlier, hospIPA hospitalizes the critical individuals corresponding to poor solutions of the problem and decreases the number of active individuals being used in the subsequent cycle. By executing this type of mechanism, hospIPA becomes capable of managing a population containing huge number of members. If hospIPA starts its optimization with a population containing a small number of members, the hospitalization mechanism can also decrease the solution diversity, but it should be noticed that hospIPA discharges some patients whose treatments conclude successfully and adjusts the number of active individuals dynamically. Also, hospIPA utilizes from a specialized treatment schema where the plasma being used for the patients is collected at the beginning of the second main stage in order to explore the neighborhood of the best solution discovered so far with the help of the the best solution of the current population and then transferred subtly to improve the exploitation characteristics of the algorithm.

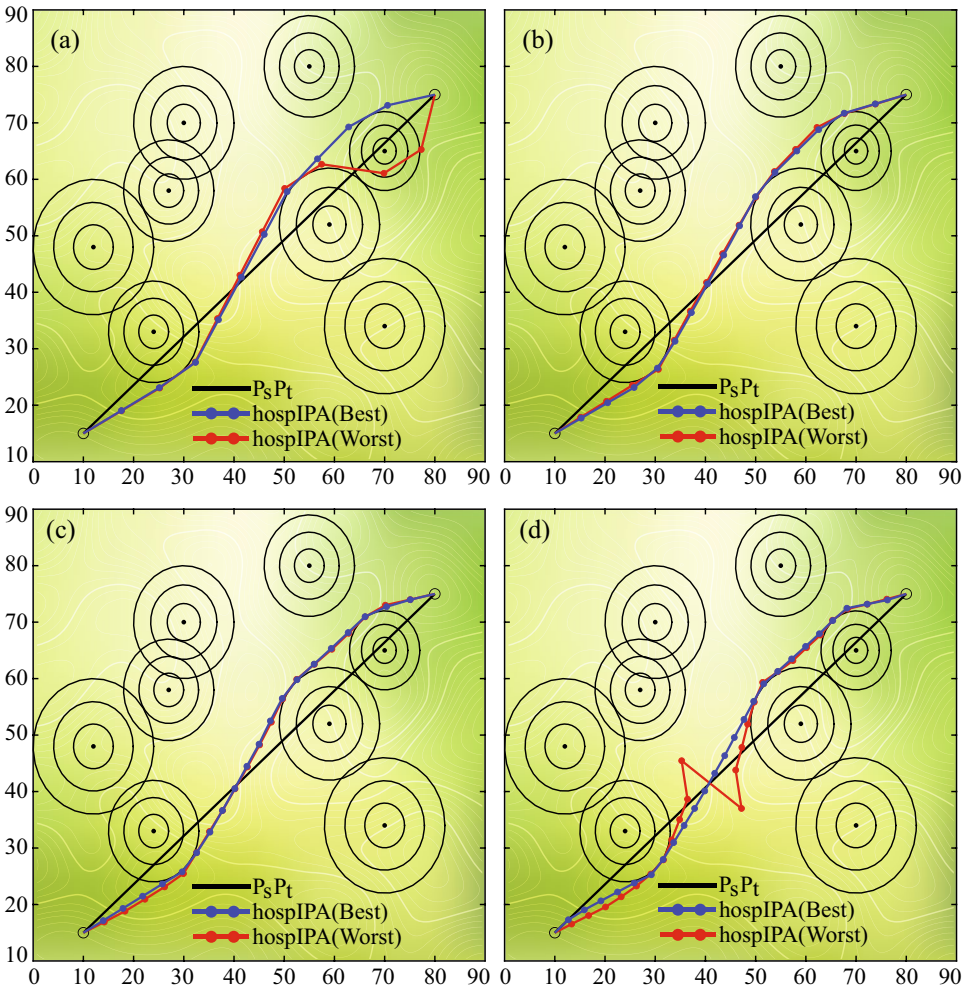

Fig. 3 The best and worst paths found by hospIPA for Scenario-1 with D equal to 10 (**a**), 15 (**b**), 20 (**c**) and 25 (**d**)

The stable performance of hospIPA was validated over the results given in Tables 3, 4, 5 for varying *PS* parameter values. However, when the results of these tables are analyzed carefully, a subtle detail about the relationship between the *PS*, battlefield scenarios and their test cases getting intrinsically difficult with the higher values of D can also be detected. While hospIPA obtains slightly better paths for the test cases of Scenario-1 and Scenario-3 by setting the *PS* to 30 compared to the paths of the same algorithm by setting the *PS* to 40, 50, 75 or 100, it requires more than 30 individuals for planning more qualified paths related with the test cases of Scenario-2. The optimal paths being calculated for the test cases of Scenario-1 and Scenario-3 contain less maneuvers than the optimal paths being calculated for the test cases of Scenario-2 and setting *PS* to a small constant such as 30 allows hospIPA utilizing from specialized operations more and finding fine-tuned paths for a UAV or UCAV. If a test case similar to the test cases of Scenario-2 includes optimal path or paths with challenging maneuvers, assigning higher values to the *PS* increases the probability of obtaining initial solutions satisfying mentioned maneuvers partially or near fully. By combining the benefits of starting optimization with a huge number of initial solutions, newly designed hospitalization mechanism and treatment schema, it is seen that hospIPA calculates better paths for the test cases of Scenario-2 when its *PS* parameter is determined as 50, 75 or 100. For a visual representation of the battlefields and the paths found by hospIPA with 30 individuals, Figs. 3, 4, 5 can be controlled.

The quality evaluation of the discovered paths by hospIPA should be made over a comparison with other metaheuristic based planners. For this purpose, a set of comparative studies between hospIPA and standard implementations of IPA, GA, MFO, SSA, PFA, SBO, SCA, GWO, AEO and improved variants of some of them such as GAPSO, ECTLBO, HSGWO-MSOS, CIPSO and NSEAEO was carried out. In order to guarantee that the comparative studies between hospIPA and other techniques are performed under the same conditions, each test case in Scenario-1, Scenario-2 and Scenario-3 was experimented 30 times by setting the population size to 30 and maximum evaluation number to 6000 and obtained results were presented in Tables 6, 7, 8.

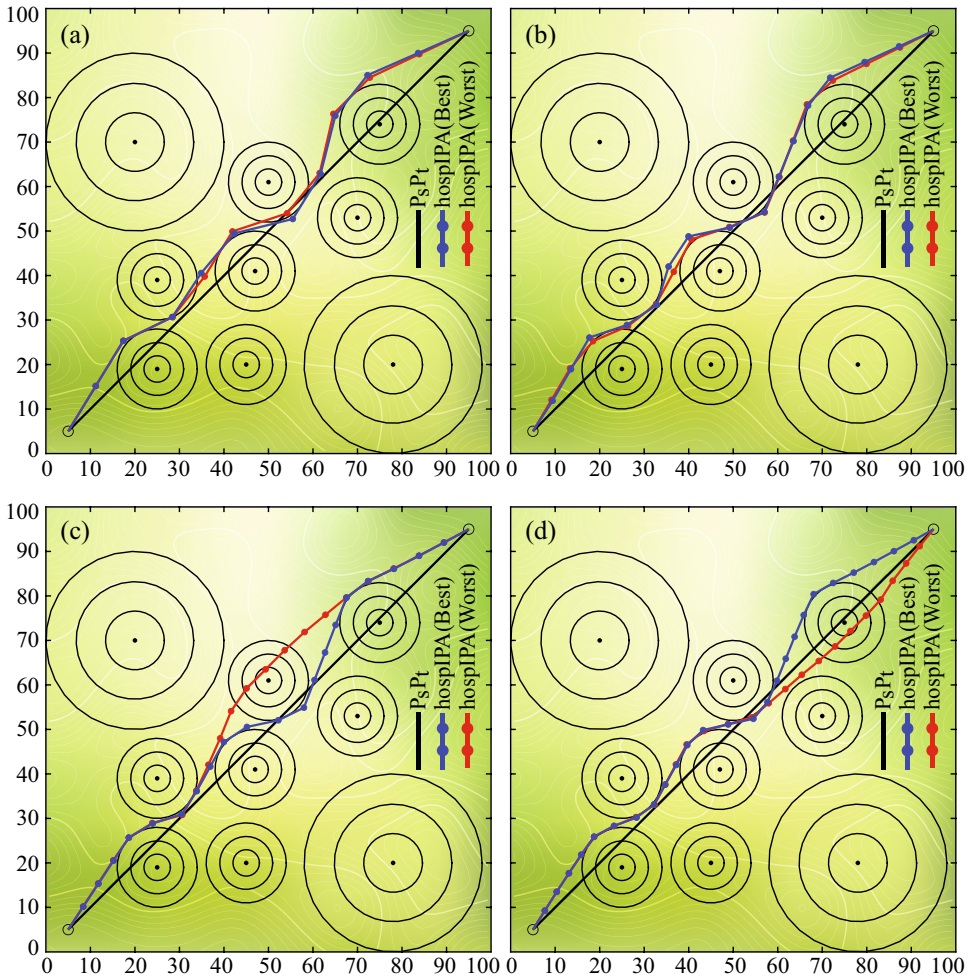

Fig. 4 The best and worst paths found by hospIPA for Scenario-2 with D equal to 10 (**a**), 15 (**b**), 20 (**c**) and 25 (**d**)

The first and foremost thing that can be extracted from the mentioned tables is the promising performance of hospIPA against its competitors. While hospIPA is determined as the best path planner among other algorithms with the average ranks equal to 1.750 for the Scenario-1 and Scenario-2, its superiority becomes more apparent for Scenario-3 and hospIPA is also determined as the best planner among other algorithms with the average rank equal to 1.000. Another important conclusion that can be extracted from Tables 6, 7, 8 is about that the performance of hospIPA increases generally compared to the other tested meta-heuristics when the number of segmentation points is chosen high enough. If the number of segmentation points is chosen high for the sensitivity, finding an optimum or near optimum path gets more difficult. However, the difficulty of path planning stemmed from the higher values of the number of segmentation points is handled successfully by hospIPA and its plasma generation and transfer schema. Because of the plasma generation depends on improving each parameter of the best solution with the help of selected donor, if the number of segmentation points is set to a relatively high value, collecting plasma that is qualifiable than the considered best solution and donor individual can be more probable. Nevertheless, it should be noticed that the cost of plasma generation in terms of consumed function evaluations rises and hospIPA can terminate without repeating its operations adequately. The mentioned drawback of hospIPA shows its effect on the path planning capabilities and hospIPA lags behind only NSE-AEO and gets ranked as the second best technique for the test cases of Scenario-1 and Scenario-2 with D equal to 25.

5.2 Planning paths for three-dimensional battlefields with hospIPA

The investigations about the path planning performance of hospIPA are continued with the experiments by using three-dimensional battlefield scenarios that are called Scenario-4 and Scenario-5. The Scenario-4 and Scenario-5 represent

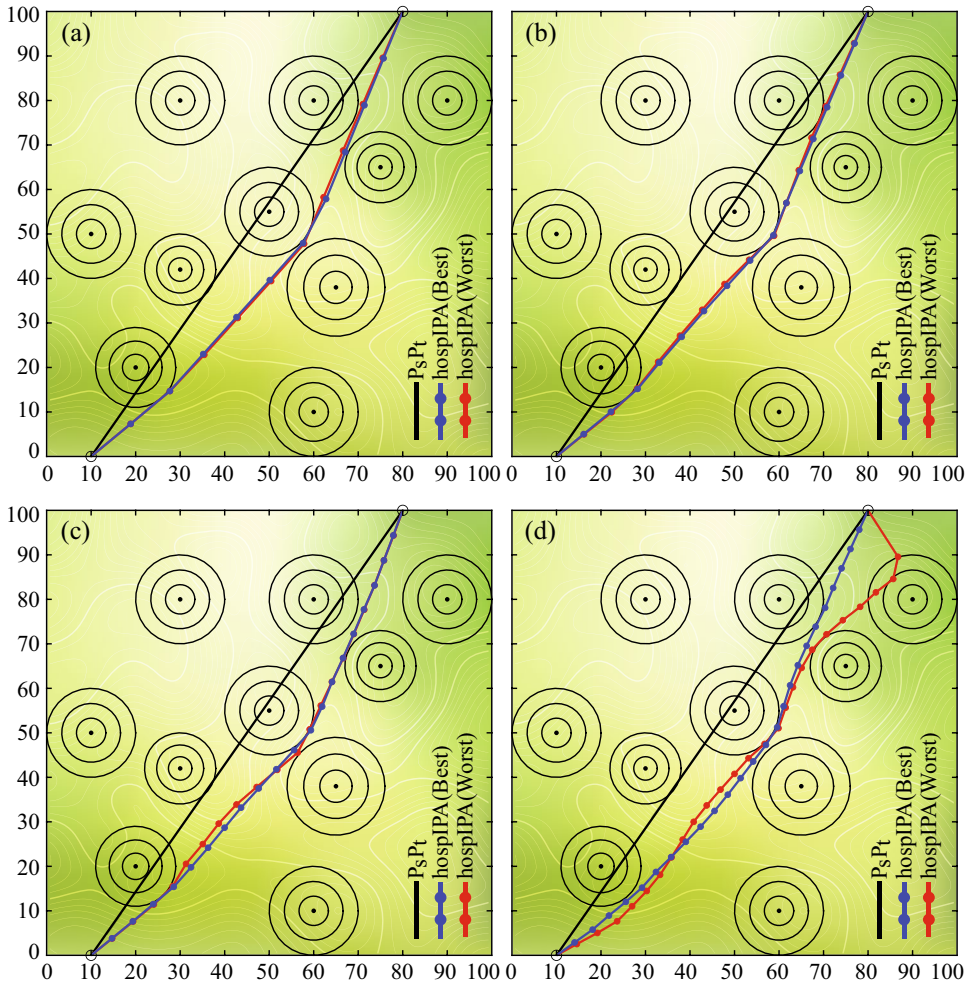

Fig. 5 The best and worst paths found by hospIPA for Scenario-3 with D equal to 10 (**a**), 15 (**b**), 20 (**c**) and 25 (**d**)

an enemy threat with a cylinder whose center, radius and height are known as detailed in Table 9 [64]. The test cases generated by assigning 10, 15, 20 and 25 constants to D for Scenario-4 and Scenario-5 were solved with hospIPA. The PS parameter of hospIPA was set to 30, 40, 50, 75 and 100 and 30 independent runs were carried out after determining maximum evaluation number as 6000. The best, worst, mean best objective function values and standard deviations of 30 independent runs were summarized in Table 10 for Scenario-4 and Table (11) for Scenario-5. From the results given in Table (10) and Table 11, it is seen that hospIPA is capable of protecting previously proven stable performance especially for the 10 and 15 dimensional cases of the considered scenarios. However, when the number of segmentation points is increased and determined as 20 and 25, hospIPA requires selection of PS parameter more carefully. The existence of z-coordinate and the higher number of segmentation points bring additional complexity to the path planning problem and hospIPA should iterate more by starting the search with relatively small PS values such as 30 or 40. If hospIPA iterates more by starting the search with relatively small PS values, the discrimination between the hospitalized and non-hospitalized individuals is carried out quickly. Moreover, repeating the plasma collection operations for each iteration allows hospIPA to explore the vicinity of the best solution and find more qualified plasma being used for the treatment of hospitalized individuals. The best and worst paths found by hospIPA with PS equal to 30 are depicted in Figs. 6, 7 for a pictorial investigations about the tested three-dimensional battlefields their and maneuver requirements.

In order to decide that whether the promising performance of hospIPA for the fixed altitude battlefield scenarios against other meta-heuristic based path planners is also achieved on the three-dimensional battlefield scenarios or not, a comparison between hospIPA and IPA, GA, MFO, SSA, PFA, SBO, SCA, GWO, AEO, GAPSO, ECTLBO, HSGWO-MSOS, CIPSO and NSEAEO was made again. Each test case in Scenario-4 and Scenario-5 was solved 30 times with hospIPA and other mentioned meta-heuristic algorithms by assigning 30 and 6000 constants to the population size and maximum evaluation number and then

Table 6 Comparison between hospIPA and other path planners for Scenario-1

D		hospIPA	IPA	GA	GAPSO	MFO	SSA	PFA	SBO	SCA	ECTLBO	HSGWO-MSOS	CIPSO	GWO	AEO	NSEAEO
10	Best	38.394	38.413	39.647	39.754	38.703	39.241	38.587	38.582	51.367	39.080	38.565	40.103	38.565	38.642	38.506
	Worst	50.915	38.467	149.678	98.436	89.657	122.790	318.996	241.127	109.966	57.912	132.095	245.441	124.542	52.378	46.526
	Mean	39.657	**38.442**	65.919	61.725	63.210	57.037	88.227	69.774	78.536	45.379	57.994	96.501	63.559	46.562	38.929
	Std	3.817	0.022	34.300	24.078	15.100	22.882	68.817	52.248	15.700	5.016	20.323	52.917	25.320	2.902	1.526
	Rank	3	1	11	8	9	6	14	12	13	4	7	15	10	5	2
15	Best	38.256	38.582	41.148	42.253	42.814	39.022	38.466	38.402	63.348	51.425	39.068	39.912	38.560	38.302	38.291
	Worst	38.335	43.773	232.634	80.672	266.532	105.756	210.693	203.260	209.946	112.515	115.202	223.591	97.052	58.778	58.042
	Mean	**38.285**	39.357	81.644	57.736	90.566	56.959	69.610	66.273	117.484	82.711	61.265	92.743	52.706	45.410	39.074
	Std	0.028	1.764	50.303	13.416	44.985	19.697	45.982	38.427	30.720	17.811	13.564	49.986	15.555	9.140	3.584
	Rank	1	3	11	7	13	6	10	9	15	12	8	14	5	4	2
20	Best	38.245	47.952	49.553	51.378	60.290	44.282	39.543	39.351	66.430	49.978	43.637	42.977	39.083	39.508	38.369
	Worst	38.350	59.512	301.088	118.726	396.167	219.033	145.096	174.852	373.008	211.499	141.557	503.643	97.515	52.182	39.606
	Mean	**38.290**	53.747	135.674	80.166	158.211	81.580	59.488	65.309	181.787	131.767	67.328	137.171	51.652	43.195	38.769
	Std	0.038	3.362	59.459	14.844	84.234	40.491	25.292	31.510	71.888	44.952	22.644	93.903	11.872	3.994	0.293
	Rank	1	5	12	9	14	10	6	7	15	11	8	13	4	3	2
25	Best	38.363	63.018	102.967	84.872	84.829	60.426	41.970	47.433	57.273	81.008	43.554	65.411	41.514	41.017	39.299
	Worst	46.183	80.786	439.336	199.769	2246.004	276.890	166.076	137.645	580.501	454.877	167.021	647.234	90.889	51.240	40.108
	Mean	41.732	76.723	173.069	122.905	282.449	122.920	77.936	68.971	252.098	187.488	81.527	205.902	50.307	42.697	**39.615**
	Std	3.852	4.735	68.499	25.532	388.776	50.105	30.544	21.100	150.970	100.894	33.750	141.196	11.722	1.995	0.182
	Rank	2	6	11	9	15	10	7	5	14	12	8	13	4	3	1
Average rank		1.750	3.750	11.250	8.250	12.750	8.000	9.250	8.250	14.250	9.750	7.750	13.750	5.750	3.750	1.750
Overall rank		1	3	12	8	13	7	10	8	15	11	6	14	5	3	1

Bold values show the better results

Table 7 Comparison between hospIPA and other path planners for Scenario-2

D		hospIPA	IPA	GA	GAPSO	MFO	SSA	PFA	SBO	SCA	ECTLBO	HSGWO -MSOS	CIPSO	GWO	AEO	NSEAEO
10	Best	57.584	57.802	61.668	61.721	59.158	59.726	57.896	58.135	88.336	60.541	57.869	59.643	61.239	58.590	56.204
	Worst	63.757	61.205	312.343	130.981	270.787	221.180	598.749	525.925	221.805	99.426	109.680	1677.440	120.087	72.425	69.813
	Mean	60.632	58.705	99.630	94.912	90.261	103.382	166.529	107.482	123.145	75.787	67.683	158.779	79.191	65.734	**57.474**
	Std	2.580	1.299	47.060	20.382	40.877	48.416	159.441	104.792	26.882	10.033	13.555	292.587	14.446	6.742	2.659
	Rank	3	2	10	9	8	11	15	12	13	6	5	14	7	4	1
15	Best	54.412	57.274	66.193	67.163	60.710	57.056	58.235	57.025	105.006	69.484	57.225	70.080	55.091	54.796	55.949
	Worst	58.646	62.937	551.464	241.538	670.571	398.873	268.370	230.585	334.849	235.419	201.640	537.313	127.408	75.777	66.180
	Mean	**56.493**	59.266	162.766	111.416	171.648	131.444	122.142	82.483	184.436	152.439	80.750	230.860	78.095	57.660	60.250
	Std	1.132	1.993	106.642	47.446	154.733	79.183	68.468	45.101	56.189	36.838	32.343	134.219	17.432	4.269	2.654
	Rank	1	3	12	8	13	10	9	7	14	11	6	15	5	2	4
20	Best	53.813	63.286	79.686	97.730	68.284	61.780	54.954	57.605	104.952	91.201	64.063	93.525	57.437	54.659	54.606
	Worst	64.771	67.483	706.929	227.695	3271.735	369.486	353.765	218.618	842.337	505.444	216.044	2238.837	182.698	84.463	58.735
	Mean	**55.374**	65.470	207.937	130.496	314.948	184.588	133.978	113.440	341.186	249.007	102.336	397.446	77.195	57.304	55.451
	Std	3.751	1.589	148.025	27.769	576.712	92.907	82.512	51.390	193.248	87.854	42.546	471.670	23.585	5.344	0.861
	Rank	1	4	11	8	13	10	9	7	14	12	6	15	5	3	2
25	Best	53.648	68.697	126.733	111.400	83.876	74.451	61.648	65.162	90.491	153.223	60.073	78.704	57.743	55.240	55.061
	Worst	63.181	78.183	1609.607	272.376	2182.624	366.400	489.869	577.207	1503.886	905.050	432.937	1980.693	110.751	63.672	56.795
	Mean	**55.984**	73.419	368.502	160.852	352.786	168.946	175.648	162.158	455.569	495.212	136.757	460.333	71.813	56.626	55.922
	Std	2.722	3.364	285.403	33.317	370.204	67.626	107.691	107.937	283.937	211.767	85.818	401.653	14.139	1.532	0.433
	Rank	2	5	12	7	11	9	10	8	13	15	6	14	4	3	1
Average rank		1.750	3.500	11.250	8.000	11.250	10.000	10.750	8.500	13.500	11.000	5.750	14.500	5.250	3.000	2.000
Overall rank		1	4	12	7	12	9	10	8	14	11	6	15	5	3	2

Bold values show the better results

Table 8 Comparison between hospIPA and other path planners for Scenario-3

D		hospIPA	IPA	GA	GAPSO	MFO	SSA	PFA	SBO	SCA	ECTLBO	HSGWO -MSOS	CIPSO	GWO	AEO	NSEAEO
10	Best	49.772	49.785	56.638	55.832	57.290	53.727	53.576	53.350	70.344	56.020	53.967	53.589	53.639	52.638	51.186
	Worst	49.797	49.817	424.807	466.463	186.974	140.715	499.851	330.582	270.192	100.391	159.386	454.467	290.474	62.066	78.808
	Mean	**49.777**	49.807	124.320	105.328	97.579	79.831	144.949	120.515	132.559	76.156	77.725	137.143	82.311	59.555	55.885
	Std	0.009	0.012	100.904	89.177	28.811	28.453	121.329	82.407	55.161	12.884	21.098	103.686	43.614	4.187	5.391
	Rank	1	2	12	10	9	7	15	11	13	5	6	14	8	4	3
15	Best	49.763	50.217	57.131	58.074	58.535	51.142	52.823	52.724	103.034	80.003	55.461	59.655	53.811	50.429	50.155
	Worst	49.815	50.400	239.798	146.225	333.608	206.804	423.168	230.734	472.427	193.046	236.558	381.843	160.144	220.555	54.568
	Mean	**49.779**	50.283	98.429	74.533	103.670	77.403	125.829	84.734	213.126	130.755	93.354	151.807	70.679	61.643	52.163
	Std	0.016	0.064	42.726	17.151	58.943	42.210	90.463	50.770	90.269	21.543	44.605	83.722	25.917	31.234	1.434
	Rank	1	2	10	6	11	7	12	8	15	13	9	14	5	4	3
20	Best	49.789	51.591	86.357	64.500	76.072	52.148	54.123	53.267	104.708	95.829	57.597	60.565	53.736	50.843	50.257
	Worst	49.927	54.415	397.440	261.488	1048.996	303.329	435.366	306.776	573.087	380.192	203.090	779.707	365.019	64.468	54.184
	Mean	**49.845**	52.841	180.662	96.442	236.951	91.752	150.474	113.194	258.060	215.535	108.225	224.292	82.149	54.906	52.492
	Std	0.055	1.015	83.360	35.996	217.852	53.819	96.867	58.481	119.930	59.956	38.526	154.675	57.171	2.108	1.113
	Rank	1	3	11	7	14	6	10	9	15	12	8	13	5	4	2
25	Best	49.878	60.958	118.945	95.917	96.659	57.896	55.473	51.546	83.408	113.601	59.155	100.647	54.723	53.19031	51.090
	Worst	62.445	73.486	789.659	230.781	1150.201	262.789	461.069	296.169	926.916	688.007	239.553	888.148	166.014	57.591	54.327
	Mean	**50.831**	65.097	227.699	138.983	395.133	105.547	129.069	112.060	356.789	347.054	111.295	367.574	74.745	55.191	53.648
	Std	3.158	4.119	133.016	28.790	283.375	48.219	93.393	53.088	206.941	159.762	44.611	196.039	25.892	0.916	0.649
	Rank	1	4	11	10	15	6	9	8	13	12	7	14	5	3	2
Average rank		1.000	2.750	11.000	8.250	12.250	6.500	11.500	9.000	14.000	10.500	7.500	13.750	5.750	3.750	2.500
Overall rank		1	3	11	8	13	6	12	9	15	10	7	14	5	4	2

Bold values show the better results

Table 9 Details of battlefields used for three-dimensional path planning

Sc	Threat centers	Threat radius	Threat height	Threat grade	Start-Target point
4	(23,60),(30,15),(45,27),(50,75),(60,10),(70,85),(78,62),(90,80)	11,9,10,11,10,8,8,7	120,80,140,110,130,100,144,160	8,10,8,11,8,7,4,9	(0,0,10) (100,90,75)
5	(25,15),(25,80),(39,40),(45,70),(55,10),(70,80),(75,50),(85,25)	10,10,8,9,12,7,11,11	80,60,100,120,130,140,80,90	10,8,5,11,3,6,13,4	(0,0,0) (80,75,50)

Table 10 Results of hospIPA with varying *PS* values for Scenario-4

D		*PS*				
		30	40	50	75	100
10	Best	60.396	60.391	60.387	60.393	60.380
	Worst	60.428	60.430	66.226	60.463	60.414
	Mean	60.411	60.405	61.377	60.419	**60.401**
	Std	0.012	0.016	2.205	0.027	0.016
15	Best	60.477	60.532	60.524	60.522	60.549
	Worst	69.564	61.843	69.558	61.923	60.591
	Mean	62.201	60.766	62.402	60.791	**60.567**
	Std	3.381	0.490	3.304	0.517	0.015
20	Best	60.436	60.386	60.437	60.581	60.463
	Worst	61.698	61.725	68.445	62.273	61.962
	Mean	**60.703**	60.848	62.083	61.348	61.324
	Std	0.505	0.597	2.935	0.661	0.665
25	Best	60.526	60.485	60.558	60.522	60.776
	Worst	73.029	62.008	61.855	62.970	78.807
	Mean	62.658	**60.690**	61.097	61.155	63.344
	Std	3.549	0.451	0.503	0.685	4.385

Bold values show the better results

Table 11 Results of hospIPA with varying *PS* values for Scenario-5

D		*PS*				
		30	40	50	75	100
10	Best	49.203	49.241	49.236	49.185	49.185
	Worst	52.350	49.296	49.417	49.256	49.882
	Mean	50.168	49.263	49.320	**49.216**	49.416
	Std	1.453	0.022	0.071	0.030	0.299
15	Best	49.129	49.141	49.149	49.160	49.186
	Worst	53.674	49.234	53.508	49.266	53.486
	Mean	49.905	**49.171**	50.069	49.187	49.824
	Std	1.714	0.032	1.383	0.039	1.462
20	Best	49.121	49.152	49.146	49.147	49.342
	Worst	52.523	52.604	49.526	74.710	56.957
	Mean	49.616	49.465	**49.248**	52.620	50.686
	Std	1.160	0.857	0.126	8.812	2.294
25	Best	49.092	49.098	49.342	49.283	49.582
	Worst	58.451	73.209	90.681	52.457	63.061
	Mean	**50.067**	53.968	57.025	50.163	52.808
	Std	2.308	6.702	12.980	1.199	4.409

Bold values show the better results

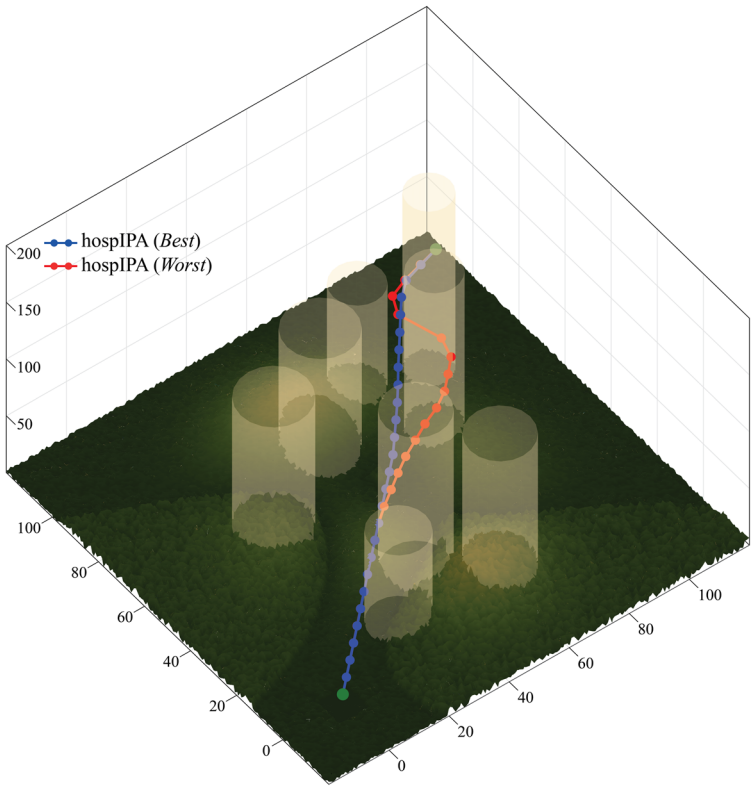

Fig. 6 The best and worst paths found by hospIPA for Scenario-4 with D equal to 25

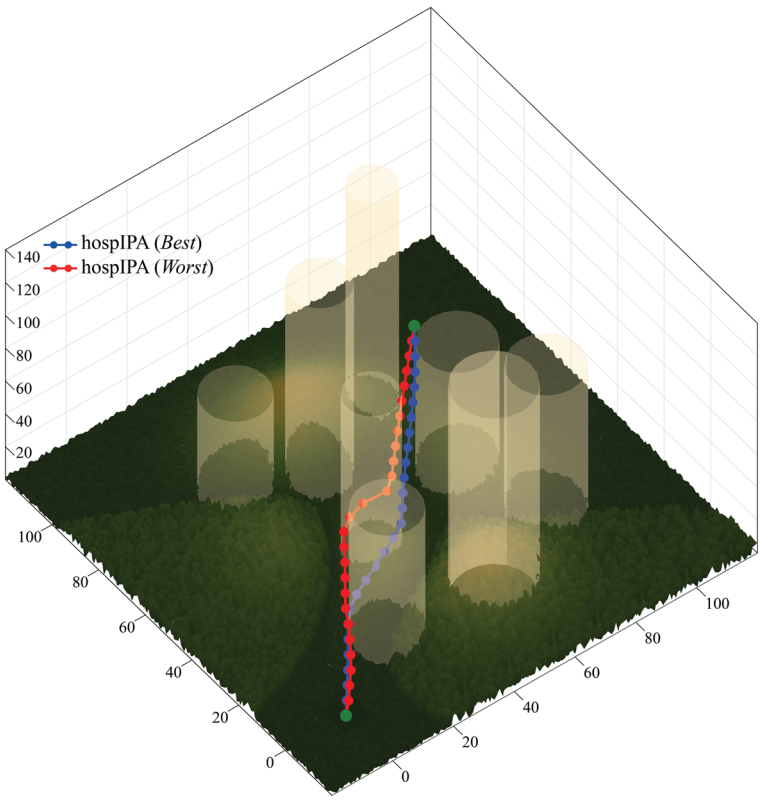

Fig. 7 The best and worst paths found by hospIPA for Scenario-5 with D equal to 25

Table 12 Comparison between hospIPA and other path planners for Scenario-4

D		hospIPA	IPA	GA	GAPSO	MFO	SSA	PFA	SBO	SCA	ECTLBO	HSGWO -MSOS	CIPSO	GWO	AEO	NSEAEO
10	Best	60.396	60.428	76.756	67.111	75.029	67.312	70.768	71.199	91.852	68.907	71.534	70.308	64.749	62.555	60.845
	Worst	60.428	60.625	158.217	102.925	132.448	129.544	198.560	119.071	190.916	124.651	154.507	241.003	158.006	95.048	68.715
	Mean	**60.411**	60.454	110.254	77.969	97.308	88.945	117.956	86.433	127.029	92.787	97.594	133.476	82.311	69.734	63.080
	Std	0.012	0.049	22.965	8.663	13.529	15.398	38.821	10.365	21.023	15.308	25.590	53.087	20.561	7.885	2.400
	Rank	1	2	12	5	10	8	13	7	14	9	11	15	6	4	3
15	Best	60.477	60.704	113.068	94.349	112.402	90.111	75.349	66.831	169.526	80.845	77.291	73.293	75.104	69.351	63.000
	Worst	69.564	62.177	400.388	225.290	229.757	249.239	321.773	166.029	387.570	228.600	233.960	470.726	188.100	116.736	71.903
	Mean	62.201	**61.545**	236.271	130.595	158.716	171.551	169.957	103.489	256.540	158.629	127.876	166.228	97.445	85.650	66.879
	Std	3.381	0.481	70.877	26.801	32.652	36.149	63.589	30.486	42.378	41.715	37.616	109.587	26.882	13.987	2.439
	Rank	2	1	14	8	10	13	12	6	15	9	7	11	5	4	3
20	Best	60.436	69.583	212.118	166.114	147.388	163.500	153.119	84.385	260.958	115.410	78.344	104.200	82.119	75.350	69.317
	Worst	61.698	83.571	645.485	242.612	513.245	370.773	605.612	277.449	528.958	383.258	287.395	494.316	208.406	177.231	82.313
	Mean	**60.703**	76.906	352.004	201.938	241.840	245.420	242.891	164.267	381.926	262.510	153.699	221.071	125.478	101.527	73.513
	Std	0.505	5.166	94.715	22.093	74.172	53.865	89.024	49.330	52.763	75.975	64.485	95.919	41.968	28.088	3.058
	Rank	1	3	14	8	10	12	11	7	15	13	6	9	5	4	2
25	Best	60.526	72.321	360.871	229.277	173.405	193.243	152.942	128.840	375.237	209.783	80.544	140.185	79.462	82.095	73.671
	Worst	73.029	124.608	815.003	386.461	653.905	483.707	459.100	411.034	1078.313	584.577	379.766	736.038	377.548	151.071	85.915
	Mean	**62.658**	93.485	470.697	288.554	305.863	314.832	283.540	244.363	560.718	366.932	137.188	359.509	135.731	90.038	77.894
	Std	3.549	19.641	109.875	35.624	101.403	62.828	64.289	60.948	145.646	100.359	75.610	133.501	60.112	12.873	2.698
	Rank	1	4	14	9	10	11	8	7	15	13	6	12	5	3	2
Average rank		1.250	2.500	13.500	7.500	10.000	11.000	11.000	6.750	14.750	11.000	7.500	11.750	5.250	3.750	2.500
Overall rank		1	2	14	7	9	10	10	6	15	10	8	13	5	4	2

Bold values show the better results

Table 13 Comparison between hospIPA and other path planners for Scenario-5

D		hospIPA	IPA	GA	GAPSO	MFO	SSA	PFA	SBO	SCA	ECTLBO	HSGWO -MSOS	CIPSO	GWO	AEO	NSEAEO
10	Best	49.203	49.039	63.953	58.587	65.548	55.416	65.060	55.742	83.149	58.946	57.964	68.905	54.410	55.725	49.521
	Worst	52.350	49.111	210.849	125.263	228.838	159.560	219.133	97.709	161.330	105.822	124.814	270.951	114.798	64.532	56.788
	Mean	50.168	**49.087**	105.116	70.071	99.140	80.013	119.546	66.524	113.678	83.595	87.189	114.524	83.284	57.357	52.663
	Std	1.453	0.032	33.480	13.037	31.422	22.023	38.797	11.571	20.770	16.218	13.503	44.396	15.829	1.826	2.634
	Rank	2	1	12	6	11	7	15	5	13	9	10	14	8	4	3
15	Best	49.129	49.828	144.123	97.437	71.380	92.825	96.046	64.484	165.075	78.945	61.897	102.656	70.663	55.740	52.141
	Worst	53.674	53.455	414.686	175.326	288.325	234.469	281.901	157.030	353.789	287.699	239.709	711.134	180.964	100.518	58.459
	Mean	**49.905**	51.717	225.811	133.764	137.688	145.071	187.006	93.061	267.275	188.067	137.615	219.629	127.641	71.299	55.378
	Std	1.714	1.311	58.365	15.323	55.691	30.044	41.949	22.135	44.212	56.586	48.832	144.745	30.012	11.9766	2.023
	Rank	1	2	14	7	9	10	11	5	15	12	8	13	6	4	3
20	Best	49.121	57.629	278.666	153.305	77.246	163.633	139.168	83.314	231.104	160.508	61.037	107.833	62.021	60.427	55.016
	Worst	52.523	78.618	590.340	269.286	570.129	298.376	712.942	207.097	661.466	466.870	386.527	1184.152	269.834	68.067	60.604
	Mean	**49.616**	69.224	356.830	203.675	245.734	220.348	260.349	144.505	378.619	311.800	176.983	410.190	161.174	63.330	57.818
	Std	1.160	8.138	77.825	33.018	109.796	37.509	103.924	34.728	98.131	92.835	87.104	302.314	54.730	1.865	1.277
	Rank	1	4	13	8	10	9	11	5	14	12	7	15	6	3	2
25	Best	49.092	67.011	381.902	260.008	107.694	184.216	179.310	146.369	312.985	201.941	68.600	113.614	65.933	63.512	57.735
	Worst	58.451	109.927	939.446	419.365	581.034	631.841	446.313	468.136	1431.116	697.660	455.734	1423.391	427.776	166.026	60.152
	Mean	**50.067**	92.290	525.074	317.795	323.891	345.003	316.295	223.264	591.812	435.614	261.152	457.170	216.622	110.245	58.848
	Std	2.308	16.163	122.076	35.055	121.414	117.991	59.176	62.073	252.107	136.389	117.664	322.862	60.112	31.083	0.652
	Rank	1	3	14	9	10	11	8	6	15	12	7	13	5	4	2
Average rank		1.250	2.500	13.250	7.500	10.000	9.250	11.250	5.250	14.250	11.250	8.000	13.750	6.250	3.750	2.500
Overall rank		1	2	13	7	10	9	11	5	15	11	8	14	6	4	2

Bold values show the better results

Table 14 Elapsed times of hospIPA and IPA with *PS* equal to 30

D		Scenario-1		Scenario-2		Scenario-3		Scenario-4		Scenario-5	
		hospIPA	IPA	hospIPA	IPA	hospIPA	IPA	hospIPA	IPA	hospIPA	IPA
10	Best	0.057	0.073	0.067	0.068	0.061	0.076	0.063	0.068	0.063	0.066
	Worst	0.083	0.103	0.116	0.128	0.103	0.101	0.085	0.100	0.097	0.092
	Mean	0.062	0.069	0.076	0.084	0.069	0.084	0.071	0.078	0.071	0.075
	Std	0.006	0.016	0.012	0.015	0.009	0.017	0.005	0.009	0.008	0.008
15	Best	0.079	0.090	0.090	0.099	0.092	0.104	0.100	0.099	0.096	0.098
	Worst	0.107	0.126	0.141	0.125	0.173	0.138	0.134	0.135	0.129	0.138
	Mean	0.085	0.097	0.101	0.109	0.116	0.115	0.106	0.110	0.104	0.109
	Std	0.006	0.008	0.012	0.006	0.018	0.009	0.007	0.011	0.008	0.011
20	Best	0.107	0.117	0.116	0.131	0.123	0.136	0.118	0.132	0.125	0.133
	Worst	0.144	0.144	0.160	0.161	0.151	0.174	0.159	0.180	0.169	0.153
	Mean	0.117	0.125	0.129	0.141	0.131	0.150	0.136	0.144	0.139	0.139
	Std	0.007	0.007	0.011	0.009	0.009	0.010	0.010	0.012	0.012	0.005
25	Best	0.136	0.145	0.144	0.161	0.147	0.171	0.162	0.160	0.158	0.162
	Worst	0.170	0.186	0.228	0.189	0.226	0.207	0.201	0.208	0.203	0.189
	Mean	0.149	0.155	0.161	0.168	0.160	0.183	0.173	0.176	0.170	0.170
	Std	0.009	0.010	0.014	0.007	0.015	0.011	0.010	0.011	0.017	0.007

their results were presented in Tables 12 and 13. When the results given in Tables 12 and 13 are investigated, it is seen that hospIPA obtains better paths than other competitors for seventy-five percent of all test cases about the three-dimensional battlefields and gets ranked as the best path planner. Moreover, for the remaining 25% of all test cases about the three-dimensional battlefields, hospIPA lags slightly behind only IP algorithm and gets ranked as the second best path planner. If the details of two test cases in which IPA performs better than hospIPA are controlled, it is understood that the number of segmentation points is equal to 10 or 15 and the difference between the mean best objective function values of the IPA based techniques is relatively small. However, if the details of the test cases in which hospIPA performs better than IPA are controlled, it is observed that the number of segmentation points for the majority of the cases is equal to 20 or 25 and the difference between the mean best objective function values of IPA based techniques is considerable high and demonstrates the effectiveness of hospIPA becoming more clearer with the growing difficulty of the path planning problem.

5.3 Execution times of hospIPA

The hospitalization mechanism used by hospIPA completely changed the interactions between the members of population. Also, it should be noticed that the operations to do with the collection of plasma in hospIPA invoke *D* different calls to the procedure responsible for calculating the objective function value of a solution and a cycle consumes *D* more evaluations. If most of the individuals are in hospital, the phase related with the distribution of infection completes quickly and considerable amount of function evaluations are spent for the plasma collection and its transfer to the receivers. Moreover, if there are more than one hospitalized individual, hospIPA does not require compute intensive

Table 15 *Sr* and *Me* values of hospIPA and IPA with *PS* equal to 30

D		Scenario-1		Scenario-2		Scenario-3		Scenario-4		Scenario-5	
		hospIPA	IPA	hospIPA	IPA	hospIPA	IPA	hospIPA	IPA	hospIPA	IPA
10	*Sr*	100.000	100.000	36.667	73.333	100.000	100.000	100.000	100.000	100.000	100.000
	Me	604.700	875.200	810.000	3598.818	84.933	375.267	159.567	1489.933	173.033	771.333
15	*Sr*	100.000	83.333	100.000	90.000	100.000	100.000	100.000	100.000	100.000	100.000
	Me	591.133	3849.560	901.533	4091.222	352.833	1548.967	483.233	2639.067	206.333	2083.500
20	*Sr*	100.000	100.000	86.667	3.333	100.000	100.000	100.000	33.333	100.000	70.000
	Me	1401.433	3834.333	1429.500	1468.000	641.333	3590.567	758.033	3963.800	508.533	4289.238
25	*Sr*	100.000	56.667	90.000	10.000	93.333	10.000	90.000	10.000	100.000	26.667
	Me	1708.600	2519.412	1848.889	752.000	1051.714	602.000	1756.630	641.333	1498.333	3685.000

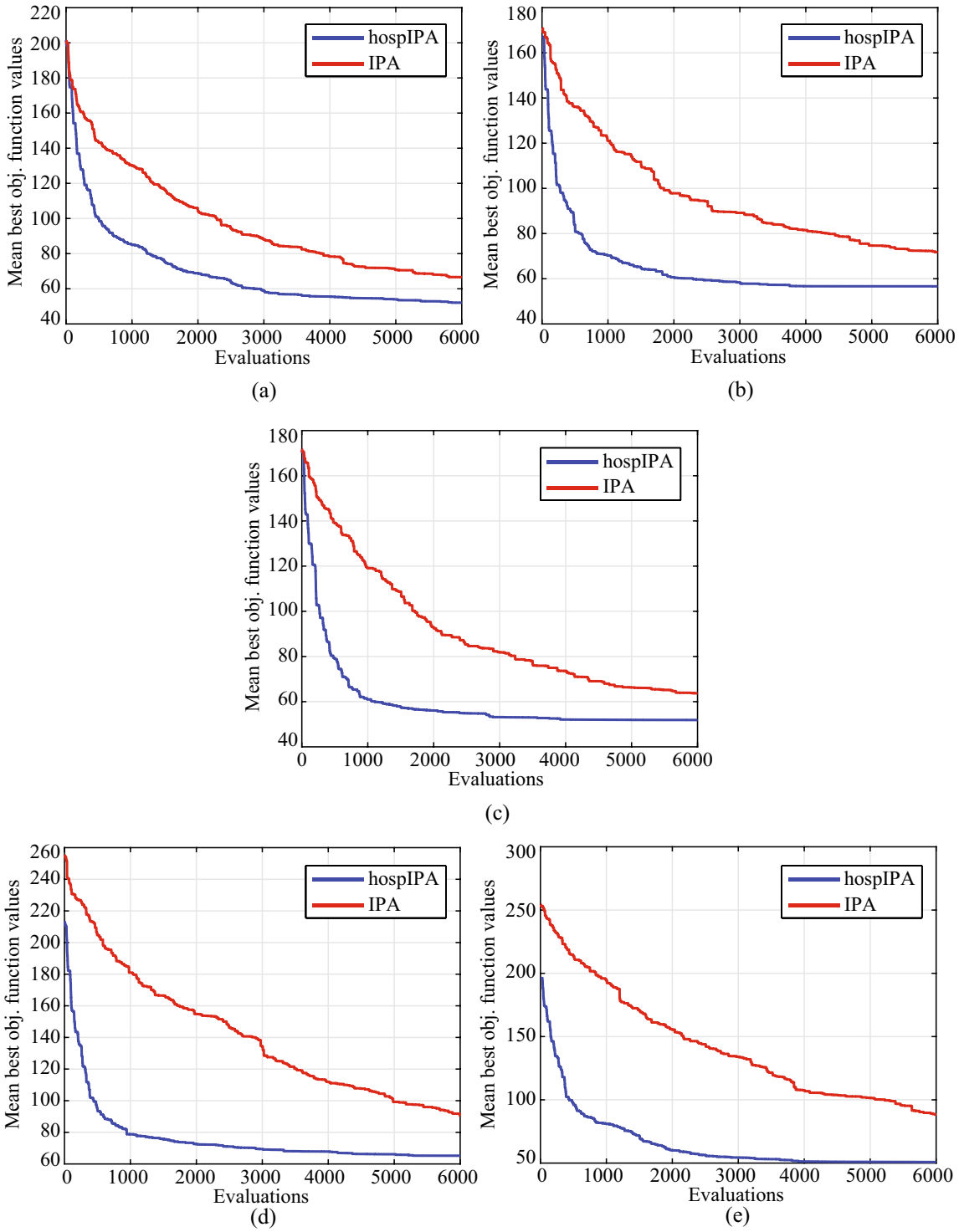

Fig. 8 Convergence curves of hospIPA and IPA for Scenario-1 (**a**), Scenario-2 (**b**), Scenario-3 (**c**), Scenario-4 (**d**) and Scenario-5 (**e**)

operations for selecting the most critical individual with the purpose of hospitalization or the best one as a donor. In order to analyze how the mentioned situations effect the execution time of hospIPA and generate a difference compared to the execution time of the standard IPA, 30 independent runs by taking *PS* and maximum evaluation number equal to 30 and 6000 were carried out. Both hospIPA and IPA were implemented in C programming language and experiments were conducted on a Fedora-34 computer with an Intel i5-10500 processor. For each run, the time elapsed until termination

Table 16 Results of the Wilcoxon signed rank test for hospIPA and IPA

D		Scenario-1	Scenario-2	Scenario-3	Scenario-4	Scenario-5
		hospIPA vs IPA	hospIPA vs IPA	hospIPA vs IPA	hospIPA vs IPA	hospIPA vs IPA
10	ρ Val	2.758e−03	6.010e−04	1.635e−06	1.501e−06	1.450e−06
	Z Val	2.993	−3.431	4.793	4.811	−4.818
	W+	87	399	0	0	465
	W−	378	66	465	465	0
	Sign	**hospIPA**	**IPA**	**hospIPA**	**hospIPA**	**IPA**
15	ρ Val	1.557e−06	1.727e−06	1.687e−06	7.243e−02	7.592e−06
	Z Val	4.803	4.782	4.787	1.796	4.476
	W+	0	0	0	145	15
	W−	465	465	465	320	450
	Sign	**hospIPA**	**hospIPA**	**hospIPA**	-	**hospIPA**
20	ρ Val	1.684e−06	1.766e−06	1.697e−06	1.702e−06	1.692e−06
	Z Val	4.788	4.778	4.786	4.785	4.787
	W+	0	0	0	0	0
	W−	465	465	465	465	465
	Sign	**hospIPA**	**hospIPA**	**hospIPA**	**hospIPA**	**hospIPA**
25	ρ Val	1.717e−06	1.729e−06	1.767e−06	1.781e−06	1.756e−06
	Z Val	4.784	4.782	4.778	4.776	4.779
	W+	0	0	0	0	0
	W−	465	465	465	465	465
	Sign	**hospIPA**	**hospIPA**	**hospIPA**	**hospIPA**	**hospIPA**

Bold values show the statistically significant algorithms

was recorded in terms of seconds and the best, worst, mean execution times and the calculated standard deviations were presented in Table 14. From the results in Table 14, it is seen that hospIPA requires less time than the standard IPA for 85 percent of all test cases. Executing a decision making approach and dynamically adjusting the number of receivers as in the mechanisms of hospIPA reduce the computational burden stemmed from the selection of the currently hospitalized individual and the donor being used for the collection of plasma.

5.4 Convergence performance and statistical analysis of hospIPA

For a numerical analysis about the convergence characteristics of a meta-heuristic algorithm, Success rate (*Sr*) and Mean evaluations (*Me*) are the two common metrics [77]. If an algorithm finds a solution whose objective function value is better than the previously determined threshold until the end of a run, the considered algorithm is assumed as successful for this run and the ratio between the number of runs for which algorithm is successful and total number of runs corresponds to the *Sr* metric. When the minimum number of evaluations required to find a solution whose objective function is better than the threshold for each successful run is recorded and then averaged, the *Me* value is obtained. The threshold was determined as 60 and 70 for the two and three-dimensional battlefield scenarios respectively and the calculated *Sr* and *Me* values of hospIPA and IPA with 30 individuals were presented in Table 15. The *Sr* and *Me* values given in Table 15 prove the superior convergence performance of hospIPA compared to the convergence performance of IPA. The *Sr* value of hospIPA is found equal or higher than the *Sr* value of IPA for 19 of all 20 test cases. Also, it should be noted that the *Me* value of hospIPA is calculated less than the *Me* value of IPA for 16 of 19 test cases in which the *Sr* value of hospIPA is equal or higher than the *Sr* value of IPA. When the remaining 3 test cases in which IPA performs better than hospIPA by considering the *Me* values are analyzed, it should be emphasized that while IPA excesses threshold only for 10 percent of all runs, hospIPA excesses the threshold for 90 percent of all runs and shows its nine times stable and consistent performance against IPA. The better convergence performance of hospIPA even though IPA outperforms its competitor by evaluating the *Me* metric for some test cases can be further validated with the convergence curves of two and three-dimensional test cases containing 25 segmentation points over Fig. 8.

Even though the positive contribution of the proposed hospitalization mechanism and treatment schema on the solving performance of hospIPA and its superiority against standard implementation of IPA can be demonstrated by checking the results of comparative studies, an appropriate statistical test should also be employed for proving the

path planning capabilities of hospIPA. The Wilcoxon signed rank test is used commonly in order to decide that one of the compared techniques is statistically better [77]. If the significance level abbreviated as ρ is less than a constant that is usually chosen as 0.05, it is accepted that the difference between two techniques is enough to generate statistical significance in favor of one of them [77]. The Wilcoxon signed rank test results for the comparison of hospIPA and IPA with 30 individuals were given in Table 16. While the Z value corresponds to the test statistics, $W+$ and $W-$ show the sum of ranks for which IPA is better than hospIPA and the sum of ranks for which hospIPA is better than IPA by considering 30 independent runs respectively in Table 16. When the ρ values calculated for the comparison between hospIPA and IPA are evaluated, it is validated that hospIPA is able to calculate paths whose qualities statistically apparent for seventeen of twenty test cases. Only for the cases with 10 segmentation points belonging to Scenario-2 and Scenario-5, the Wilcoxon signed rank test indicates that the statistical significance is in favor of IPA. As seen from the properties and number of the test cases for which hospIPA is statistically better than IPA, the newly introduced variant manages the difficulties of the paths being planned in detail by calculating the convenient values for more than 15 segmentation points.

6 Conclusion

The advantages coming with the usage of UAVs and UCAVs caused strategical changes on the military projections of nations and immense budgets were released in order to improve the performance and task success of these modern vehicles. Because of the direct impact on the performance and task success of a UAV or UCAV system, solving a problem called path planning optimally by considering the enemy threats, fuel or battery consumption and some limitations about the maneuverability became more important. Immune Plasma algorithm (IP algorithm or IPA) has been introduced recently to the literature of intelligent optimization techniques. In this study, IP algorithm was powered with a hospitalization mechanism that generates a hospital, fills it with the critical patients corresponding to the poor solutions of the population and decides who will be discharged after the plasma transfer. Moreover, the existing treatment schema was remodeled in a manner that the plasma being transferred will be gathered over the best solution found so far and the donor chosen from the population. The new IPA variant supported with the mentioned hospitalization mechanism and treatment schema that together remove the requirement of *NoR* and *NoD* parameters was named the hospital IPA (hospIPA) and employed as a UAV or UCAV path planner.

The paths planned by hospIPA for two and three-dimensional battlefields were compared with the paths planned by a set of well-known meta-heuristics and some of their variants. The experimental studies allowed to conclude that hospIPA is more qualified path planner than the tested algorithms. While hospIPA outperforms the considered path planners for the fourteen of all twenty test cases, it is ranked as the second or third best solver for the six remaining cases and still proves its competitive performance. The hospitalization mechanism selecting the poor solutions and quarantining them in a hospital helps hospIPA to explore the vicinity of the qualified solutions steadily. Furthermore, if the newly designed treatment schema that models the collection of the plasma by exploiting the best solution found so far does not give a substantial contribution to a hospitalized individual or it is not enough to discharging, hospitalization is continued for the considered individual. As an expected result of the mentioned decision, the evaluations are consumed more effectively for the non-hospitalized individuals or qualified solutions in the population. The future works can be devoted to the researches about the IPA based path planners for which the hospitalization mechanism and treatment schema are selected adaptively by considering the properties of the population. Also, the performance of the hospIPA or similar variants can be investigated by planning paths for multiple UAV or UCAV systems in a battlefield with static and dynamic enemy threats, non-flight zones and its challenging cases containing relatively high number of segmentation points.

Acknowledgements The author acknowledges to anonymous reviewer for their thoughtful suggestions and comments.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK). This study was not funded by any organisation.

Availability of data and materials Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

Conflict of interest The author has no competing interests to declare that are relevant to the content of this article.

Ethics approval and consent to participate This article does not contain any studies with human participants or animals performed by any of the authors.

Consent for publication This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Aggarwal S, Kumar N (2020) Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges. Comput Commun 149:270–299. https://doi.org/10.1016/j.comcom.2019.10.014
2. Wu Y (2021) A survey on population-based meta-heuristic algorithms for motion planning of aircraft. Swarm Evolut Comput 62:100844. https://doi.org/10.1016/j.swevo.2021.100844
3. Wen N, Su X, Ma P, Zhao L, Zhang Y (2017) Online uav path planning in uncertain and hostile environments. Int J Mach Learn Cybern 8:469–487. https://doi.org/10.1007/s13042-015-0339-4
4. Ming Z, Lingling Z, Xiaohong S, Peijun M, Yanhang Z (2017) Improved discrete mapping differential evolution for multi-unmanned aerial vehicles cooperative multi-targets assignment under unified model. Int J Mach Learn Cybern 8:765–780. https://doi.org/10.1007/s13042-015-0364-3
5. Cuevas E, Trujillo A, Navarro MA, Diaz P (2020) Comparison of recent metaheuristic algorithms for shape detection in images. Int J Comput Intell Syst 13(1):1059–1071. https://doi.org/10.2991/ijcis.d.200729.001
6. Cuevas E, Rodríguez A, Perez M, Murillo-Olmos J, Morales-Castañeda B, Alejo-Reyes A, Sarkar R (2023) Optimal evaluation of re-opening policies for covid-19 through the use of metaheuristic schemes. Appl Math Model 121:506–523. https://doi.org/10.1016/j.apm.2023.05.012
7. Aslan S, Demirci S (2020) Immune plasma algorithm: a novel meta-heuristic for optimization problems. IEEE Access 8:220227–220245. https://doi.org/10.1109/ACCESS.2020.3043174
8. Aslan S, Demirci S (2022) An improved immune plasma algorithm with a regional pandemic restriction. Signal Image Video Process 16(8):2093–2101. https://doi.org/10.1007/s11760-022-02171-w
9. Kisa M, Demirci S, Arslan S, Aslan S (2021) Solving channel assignment problem in cognitive radio networks with immune plasma algorithm. In: 2021 6th International Conference on Computer Science and Engineering (UBMK), pp. 818–822. IEEE. https://doi.org/10.1109/UBMK52708.2021.9558950
10. Tasdemir A, Demirci S, Aslan S (2022) Performance investigation of immune plasma algorithm on solving wireless sensor deployment problem. In: 2022 9th International Conference on Electrical and Electronics Engineering (ICEEE), pp. 296–300. IEEE. https://doi.org/10.1109/ICEEE55327.2022.9772539
11. Ajay V, Nesasudha M (2022) Detection of attackers in cognitive radio network using optimized neural networks. Intell Autom Soft Comput 34(1):193–204. https://doi.org/10.32604/iasc.2022.024839
12. Aslan S, Erkin T (2023) An immune plasma algorithm based approach for ucav path planning. J King Saud Univ Comput Inform Sci 35(1):56–69. https://doi.org/10.1016/j.jksuci.2022.06.004
13. Duan H, Yu Y, Zhou R (2008) Ucav path planning based on ant colony optimization and satisficing decision algorithm. In: 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), pp. 957–962. IEEE. https://doi.org/10.1109/CEC.2008.4630912
14. Duan H, Yu Y, Zhang X, Shao S (2010) Three-dimension path planning for ucav using hybrid meta-heuristic aco-de algorithm. Simul Model Pract Theory 18(8):1104–1115. https://doi.org/10.1016/j.simpat.2009.10.006
15. Ma Q, Lei X (2009) Application of improved particle swarm optimization algorithm in ucav path planning. In: International Conference on Artificial Intelligence and Computational Intelligence, pp. 206–214. Springer. https://doi.org/10.1007/978-3-642-05253-8-23
16. Xu C, Duan H, Liu F (2010) Chaotic artificial bee colony approach to uninhabited combat air vehicle (ucav) path planning. Aerosp Sci Technol 14(8):535–541. https://doi.org/10.1016/j.ast.2010.04.008
17. Zhang Y, Wu L, Wang S (2011) Ucav path planning based on fscabc. Inform Int Interdiscip J 14(3):687–692
18. Zhang Y, Wu L, Wang S (2013) Ucav path planning by fitness-scaling adaptive chaotic particle swarm optimization. Math Prob Eng 2013. https://doi.org/10.1155/2013/705238
19. Li P, Duan H (2012) Path planning of unmanned aerial vehicle based on improved gravitational search algorithm. Sci China Technol Sci 55(10):2712–2719. https://doi.org/10.1007/s11431-012-4890-x
20. Fu Z-F (2012) Path planning of ucav based on a modified geesepso algorithm. In: International Conference on Intelligent Computing, pp. 471–478. https://doi.org/10.1007/978-3-642-31576-3-60
21. Wang G-G, Guo L, Duan H, Liu L, Wang H et al (2012) A modified firefly algorithm for ucav path planning. Int J Hybrid Inform Technol 5(3):123–144
22. Wang G-G, Guo L, Duan H, Wang H, Liu,L, Shao M (2012) A hybrid metaheuristic de/cs algorithm for ucav three-dimension path planning. Sci World J 2012. https://doi.org/10.1100/2012/583973
23. Wang G-G, Guo L, Duan H, Liu L, Wang H (2012) A bat algorithm with mutation for ucav path planning. Sci World J 2012. https://doi.org/10.1100/2012/418946
24. Wang G-G, Chu HE, Mirjalili S (2016) Three-dimensional path planning for ucav using an improved bat algorithm. Aerosp Sci Technol 49:231–238. https://doi.org/10.1016/j.ast.2015.11.040
25. Liu C, Gao Z, Zhao W (2012) A new path planning method based on firefly algorithm. In: 2012 Fifth International Joint Conference on Computational Sciences and Optimization, pp. 775–778. IEEE. https://doi.org/10.1109/CSO.2012.174
26. Zhu W, Duan H (2014) Chaotic predator-prey biogeography-based optimization approach for ucav path planning. Aerosp Sci Technol 32(1):153–161. https://doi.org/10.1016/j.ast.2013.11.003
27. Heidari A, Abbaspour R (2014) Improved black hole algorithm for efficient low observable ucav path planning in constrained aerospace. Adv Comput Sci Int J 3(3):87–92
28. Tang Z, Zhou Y (2015) A glowworm swarm optimization algorithm for uninhabited combat air vehicle path planning. J Intell Syst 24(1):69–83. https://doi.org/10.1515/jisys-2013-0066
29. Yu G, Song H, Gao J (2014) Unmanned aerial vehicle path planning based on tlbo algorithm. Int J Smart Sens Intell Syst 7(3):1310–1325. https://doi.org/10.21307/ijssis-2017-707
30. Zhang X, Duan H (2015) An improved constrained differential evolution algorithm for unmanned aerial vehicle global route planning. Appl Soft Comput 26:270–284. https://doi.org/10.1016/j.asoc.2014.09.046
31. Zhou Q, Zhou Y, Chen X (2014) A wolf colony search algorithm based on the complex method for uninhabited combat air vehicle path planning. Int J Hybrid Inform Technol 7(1):183–200. https://doi.org/10.14257/ijhit.2014.7.1.15

32. Duan H, Qiao P (2014) Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning. Int J Intell Comput Cybern 7(1):24–37. https://doi.org/10.1108/IJICC-02-2014-0005
33. Li B, Gong L-G, Yang W-L (2014) An improved artificial bee colony algorithm based on balance-evolution strategy for unmanned combat aerial vehicle path planning. Sci World J 2014. https://doi.org/10.1155/2014/232704
34. Zhang B, Duan H (2015) Three-dimensional path planning for uninhabited combat aerial vehicle based on predator-prey pigeon-inspired optimization in dynamic environment. IEEE/ACM Trans Comput Biol Bioinform 14(1):97–107. https://doi.org/10.1109/TCBB.2015.2443789
35. Chen Y, Yu J, Mei Y, Wang Y, Su X (2016) Modified central force optimization (mcfo) algorithm for 3d uav path planning. Neurocomputing 171:878–888. https://doi.org/10.1016/j.neucom.2015.07.044
36. Zhou Y, Bao Z, Wang R, Qiao S, Zhou Y (2015) Quantum wind driven optimization for unmanned combat air vehicle path planning. Appl Sci 5(4):1457–1483. https://doi.org/10.3390/app5041457
37. Zhang S, Zhou Y, Li Z, Pan W (2016) Grey wolf optimizer for unmanned combat aerial vehicle path planning. Adv Eng Softw 99:121–136. https://doi.org/10.1016/j.advengsoft.2016.05.015
38. Liu Y, Zhang X, Guan X, Delahaye D (2016) Adaptive sensitivity decision based path planning algorithm for unmanned aerial vehicle with improved particle swarm optimization. Aerosp Sci Technol 58:92–102. https://doi.org/10.1016/j.ast.2016.08.017
39. Liu Y, Zhang X, Zhang Y, Guan X (2019) Collision free 4d path planning for multiple uavs based on spatial refined voting mechanism and pso approach. Chin J Aeron 32(6):1504–1519. https://doi.org/10.1016/j.cja.2019.03.026
40. Luo Q, Li L, Zhou Y (2017) A quantum encoding bat algorithm for uninhabited combat aerial vehicle path planning. Int J Innov Comput Appl 8(3):182–193. https://doi.org/10.1504/IJICA.2017.086642
41. Zhang Q, Wang R, Yang J, Ding K, Li Y, Hu J (2018) Modified collective decision optimization algorithm with application in trajectory planning of uav. Appl Intell 48(8):2328–2354. https://doi.org/10.1007/s10489-017-1082-1
42. Alihodzic A, Tuba E, Capor-Hrosik R, Dolicanin E, Tuba M (2017) Unmanned aerial vehicle path planning problem by adjusted elephant herding optimization. In: 2017 25th Telecommunication Forum (Telfor), pp. 1–4. IEEE. https://doi.org/10.1109/TELFOR.2017.8249468
43. Alihodzic A, Hasic D, Selmanovic E (2018) An effective guided fireworks algorithm for solving ucav path planning problem. In: International Conference on Numerical Methods and Applications, pp. 29–38. https://doi.org/10.1007/978-3-030-10692-8-3
44. Miao F, Zhou Y, Luo Q (2019) A modified symbiotic organisms search algorithm for unmanned combat aerial vehicle route planning problem. J Oper Res Soc 70(1):21–52. https://doi.org/10.1080/01605682.2017.1418151
45. Dolicanin E, Fetahovic I, Tuba E, Capor-Hrosik R, Tuba M (2018) Unmanned combat aerial vehicle path planning by brain storm optimization algorithm. Stud Inform Control 27(1):15–24
46. Pan J-S, Liu J-L, Hsiung S-C (2019) Chaotic cuckoo search algorithm for solving unmanned combat aerial vehicle path planning problems. In: Proceedings of the 2019 11th International Conference on Machine Learning and Computing, pp. 224–230. https://doi.org/10.1145/3318299.3318310
47. Pan J-S, Liu J-L, Liu E-J (2019) Improved whale optimization algorithm and its application to ucav path planning problem. In: International Conference on Genetic and Evolutionary Computing, pp. 37–47. https://doi.org/10.1007/978-981-13-5841-8_5
48. Pan J-S, Liu N, Chu S-C (2020) A hybrid differential evolution algorithm and its application in unmanned combat aerial vehicle path planning. IEEE Access 8:17691–17712. https://doi.org/10.1109/ACCESS.2020.2968119
49. Lin N, Tang J, Li X, Zhao L (2019) A novel improved bat algorithm in uav path planning. J Comput Mater Contin 61:323–344. https://doi.org/10.32604/cmc.2019.05674
50. Qu C, Gai W, Zhong M, Zhang J (2020) A novel reinforcement learning based grey wolf optimizer algorithm for unmanned aerial vehicles (uavs) path planning. Appl Soft Comput 89:106099. https://doi.org/10.1016/j.asoc.2020.106099
51. Qu C, Gai W, Zhang J, Zhong M (2020) A novel hybrid grey wolf optimizer algorithm for unmanned aerial vehicle (uav) path planning. Knowl Based Syst 194:105530. https://doi.org/10.1016/j.knosys.2020.105530
52. Yi J-H, Lu M, Zhao X-J (2020) Quantum inspired monarch butterfly optimisation for ucav path planning navigation problem. Int J Bio-Inspired Comput 15(2):75–89. https://doi.org/10.1504/ijbic.2020.106428
53. Wu C, Huang X, Luo Y, Leng S (2020) An improved fast convergent artificial bee colony algorithm for unmanned aerial vehicle path planning in battlefield environment. In: 2020 IEEE 16th International Conference on Control & Automation (ICCA), pp. 360–365. IEEE. https://doi.org/10.1109/ICCA51439.2020.9264555
54. Chen Y, Pi D, Xu Y (2021) Neighborhood global learning based flower pollination algorithm and its application to unmanned aerial vehicle path planning. Expert Syst Appl 170:114505. https://doi.org/10.1016/j.eswa.2020.114505
55. Zhu H, Wang Y, Ma Z, Li X (2021) A comparative study of swarm intelligence algorithms for ucav path-planning problems. Mathematics 9(2):171. https://doi.org/10.3390/math9020171
56. Zhu H, Wang Y, Li X (2022) Ucav path planning for avoiding obstacles using cooperative co-evolution spider monkey optimization. Knowl Based Syst 246:108713. https://doi.org/10.1016/j.knosys.2022.108713
57. Zhou X, Gao F, Fang X, Lan Z (2021) Improved bat algorithm for uav path planning in three-dimensional space. IEEE Access 9:20100–20116. https://doi.org/10.1109/ACCESS.2021.3054179
58. Wu P, Li T, Song G (2020) Ucav path planning based on improved chaotic particle swarm optimization. In: 2020 Chinese Automation Congress (CAC), pp. 1069–1073. IEEE. https://doi.org/10.1109/CAC51589.2020.9326556
59. Xu H, Jiang S, Zhang A (2021) Path planning for unmanned aerial vehicle using a mix-strategy-based gravitational search algorithm. IEEE Access 9:57033–57045. https://doi.org/10.1109/ACCESS.2021.3072796
60. Jiang W, Lyu Y, Li Y, Guo Y, Zhang W (2022) Uav path planning and collision avoidance in 3d environments based on pompd and improved grey wolf optimizer. Aerosp Sci Technol 121:107314. https://doi.org/10.1016/j.ast.2021.107314
61. Jarray R, Al-Dhaifallah M, Rezk H, Bouallègue S (2022) Parallel cooperative coevolutionary grey wolf optimizer for path planning problem of unmanned aerial vehicles. Sensors 22(5):1826. https://doi.org/10.3390/s22051826
62. Du N, Zhou Y, Deng W, Luo Q (2022) Improved chimp optimization algorithm for three-dimensional path planning problem. Multimed Tools Appl 81(19):27397–27422. https://doi.org/10.1007/s11042-022-12882-4
63. Wang X, Pan J-S, Yang Q, Kong L, Snášel V, Chu S-C (2022) Modified mayfly algorithm for uav path planning. Drones 6(5):134. https://doi.org/10.3390/drones6050134
64. Niu Y, Yan X, Wang Y, Niu Y (2022) An adaptive neighborhood-based search enhanced artificial ecosystem optimizer for ucav path planning. Expert Syst Appl 208:118047. https://doi.org/10.1016/j.eswa.2022.118047

65. Niu Y, Yan X, Wang Y, Niu Y (2023) Three-dimensional ucav path planning using a novel modified artificial ecosystem optimizer. Expert Syst Appl 217:119499. https://doi.org/10.1016/j.eswa.2022.119499
66. Jia Y, Qu L, Li X (2022) A double-layer coding model with a rotation-based particle swarm algorithm for unmanned combat aerial vehicle path planning. Eng Appl Artif Intell 116:105410. https://doi.org/10.1016/j.engappai.2022.105410
67. Zhang C, Zhou W, Qin W, Tang W (2023) A novel uav path planning approach: heuristic crossing search and rescue optimization algorithm. Expert Syst Appl 215:119243. https://doi.org/10.1016/j.eswa.2022.119243
68. Ait-Saadi A, Meraihi Y, Soukane A, Ramdane-Cherif A, Gabis AB (2022) A novel hybrid chaotic aquila optimization algorithm with simulated annealing for unmanned aerial vehicles path planning. Comput Elect Eng 104:108461. https://doi.org/10.1016/j.compeleceng.2022.108461
69. Yu X, Jiang N, Wang X, Li M (2023) A hybrid algorithm based on grey wolf optimizer and differential evolution for uav path planning. Expert Syst Appl 215:119327. https://doi.org/10.1016/j.eswa.2022.119327
70. Chowdhury A, De D (2023) Rgso-uav: reverse glowworm swarm optimization inspired uav path-planning in a 3d dynamic environment. Ad Hoc Netw 140:103068. https://doi.org/10.1016/j.adhoc.2022.103068
71. Chen B, Yang J, Zhang H, Yang M (2023) An improved spherical vector and truncated mean stabilization based bat algorithm for uav path planning. IEEE Access 11:2396–2409. https://doi.org/10.1109/ACCESS.2023.3234057
72. Huang C, Zhou X, Ran X, Wang J, Chen H, Deng W (2023) Adaptive cylinder vector particle swarm optimization with differential evolution for uav path planning. Eng Appl Artif Intell 121:105942. https://doi.org/10.1016/j.engappai.2023.105942
73. Hu G, Zhong J, Wei G (2023) Sachba_pdn: Modified honey badger algorithm with multi-strategy for uav path planning. Expert Syst Appl 223:119941. https://doi.org/10.1016/j.eswa.2023.119941
74. Parkin J, Cohen B (2001) An overview of the immune system. The Lancet 357(9270):1777–1789. https://doi.org/10.1016/S0140-6736(00)04904-7
75. Hung IF, To KK, Lee C-K, Lee K-L, Chan K, Yan W-W, Liu R, Watt C-L, Chan W-M, Lai K-Y (2011) Convalescent plasma treatment reduced mortality in patients with severe pandemic influenza a (h1n1) 2009 virus infection. Clin Infect Dis 52(4):447–456. https://doi.org/10.1093/cid/ciq106
76. Shen C, Wang Z, Zhao F, Yang Y, Li J, Yuan J, Wang F, Li D, Yang M, Xing L (2020) Treatment of 5 critically ill patients with covid-19 with convalescent plasma. JAMA 323(16):1582–1589. https://doi.org/10.1001/jama.2020.4783
77. Aslan S (2020) A comparative study between artificial bee colony (abc) algorithm and its variants on big data optimization. Mem Comput 12(2):129–150. https://doi.org/10.1007/s12293-020-00298-2