THE PUBLISHING HOUSE
OF THE ROMANIAN ACADEMY

PROCEEDINGS OF THE ROMANIAN ACADEMY, Series A,
Volume 23, Number 2/2022, pp. 209–218

CROSSOVER RECOMBINATION-BASED GLOBAL-BEST BRAIN STORM OPTIMIZATION ALGORITHM FOR UAV PATH PLANNING

Qian ZHOU[1,2], Shesheng GAO[2], Boyang QU[3], Xun GAO[2], Yongming ZHONG[4]

[1] Research and Development Institute of Northwestern Polytechnical University in Shenzhen, Shenzhen 518057, China
[2] Northwestern Polytechnical University, School of Automatics, Xi'an 710072, China
[3] Zhongyuan University of Technology, School of Electrical and Information Engineering, Zhengzhou 450007, China
[4] RMIT University, School of Engineering, Melbourne 3083, Australia
Corresponding author: Qian ZHOU, E-mail: zqjmbt@sina.com

Abstract. Path planning of the UAV is one of the complex optimization problems, due to the model complexity and a high number of constraints. In addition, the flyability of path is also a requirement for 3D UAV path planning in practical environment. Evolutionary algorithms are effective solutions to solve complex optimization problems with multiple constraints. Regarding the local adjustment characteristic of cubic B-spline curves and crossover recombination in differential evolution algorithm, we design and implement a crossover recombination based global-best brain storm optimization (GBSO) algorithm to solve multi-constraints 3D path planning problem with considering the continuous curvature of path. The cost function is formulated includes the safety, economy and flyability, where the characteristic polygon vertices of a cubic B-spline curve representing the path are taken as the optimization variables. Simulation results and comparison analysis demonstrate that the proposed method has a better performance than GBSO, SHADE and other compared algorithms for UAV path planning.

***Key words**:* unmanned aerial vehicle (UAV), path planning, global-best brain storm optimization algorithm, crossover recombination.

1. INTRODUCTION

In numerous areas, autonomous systems are replacing manual operations to improve the efficiency and cost-effectiveness. Due to the advantages of simplicity, low cost, safety, flexibility and reliability, unmanned aerial vehicles (UAVs) are widely used in military and civilian applications [1-3]. Especially in dangerous, remote and harsh environments, man-machine systems must be replaced by UAVs. The UAV path planning is a fundamental issue in UAV maneuvering and control, as it directly affects the performance of UAV autonomy [4]. It aims to find the optimal or near-optimal solution in 3D space, which meets the mission requirements under various environment and flight conditions [5].

A significant amount of research efforts have focused on finding a collision free and optimal path for UAV to avoid risks. The traditional methods, such as visibility graph, rapidly-exploring random tree, probabilistic roadmap, dijsktra's algorithm, A* algorithm, D* algorithm, etc. These methods provide solutions for generating optimal paths in the geometrical, but the paths without considering the motion constraints of UAV and cannot be used in practical applications [6]. Moreover, these methods are relatively slow speed and execution [7].

For solving optimization problems, the dynamic programming (DP) is one traditional solution [8]. However, the DP depends heavily on the specificity of problem and will be slow with the gradual increase scale of problem. With development of science, the meta-heuristic algorithms have received great attention. The discrete and improved bat algorithm (DaIBA) was proposed for solving vehicle routing problem (VRP) with asymmetric variable costs, forbidden roads and cost constraints [9]. The [10] provided us a good solution to solve the vehicle tours with time window (VRPTW) problem by adapting meta-heuristic methods. Finding the optimal path is a nondeterministic polynomial (NP) time complete problem, where the complexity increases rapidly with the dimension size. As we all know, the meta-heuristic algorithms are effective solutions to solve the NP-complete problems [11, 12]. These include the genetic algorithm (GA), particle swarm optimization, differential evolution (DE), ant colony optimization, design GA, evolutionary algorithm and combination of these algorithms. The studies show that enhances the local search and exploitation ability of algorithm are beneficial for path planning in complex environment.

In addition to the traditional heuristic algorithms, advanced nature-inspired algorithms have also received great attention in the field of UAV path planning. One typical example is the brain storm optimization (BSO), which has two major operators: convergent operator and divergent operator [13]. It mainly draws on the core ideas of the brainstorming process: judgment, assumptions and cross-reference. Through a large number of assumptions and conjectures, there is a great possibility that an excellent solution can be obtained in the end. However, BSO suffers from the problem of trapping in local optimum and has a slow convergent speed [14]. Research efforts have been dedicated on improvement of BSO in terms of clustering, creation and selection strategies [15]. One of them is the global-best brain storm optimization (GBSO), which is an algorithm to improve the BSO performance using multiple modifications [16]. The GBSO adopts fitness-based grouping, per-variable updating and the global-best guidance concept originally proposed in PSO to improve the performance of the original BSO. It is worth mentioning that the per-variable updating generates new individual one problem variable at a time rather than all problem variables in one step. The per-variable updating can enhance the local search ability of algorithm, which has the nature advantage for solving UAV path planning problem. Based the fact that the object of crossover recombination is also a single variable, the crossover recombination is suitable for GBSO. To improve the local search and exploitation ability of GBSO, the crossover recombination in DE [17] is borrowed.

Continuous path curvature is also a basic requirement for UAV path planning. Comparing to the other existing methods for path representation, such as the Dubins curve Dubins path, Bezier curves and Pythagorean Hodograph curves [18], the cubic B-spline curve has incomparable advantages for path representation. It can achieve the curvature continuity with the lowest (cubic) degree. More importantly, it enables the local adjustment of a curve. When one control point in the B-spline curve is adjusted, only the segment associated with this control point of the curve is changed, while the other segments of the curve remains unchanged. This characteristic of local shape modification provided by the cubic B-spline makes it easier for the path to avoid obstacles and risks and meet the requirements of the physical characteristics of UAV flight.

In this work, our motivation is to solve 3D UAV path planning in complex environment. The main contributions of our paper can be summarized as follows:

(1) The 3D UAV path planning is formulated as a constrained complex optimization problem with consideration of the kinematics and dynamic constraints for UAV. The cost function is formulated includes the safety and economy, where the characteristic polygon vertices of cubic B-spline curve representing the path are taken as the optimization variables.

(2) A crossover recombination-based global-best brain storm optimization (GBSO-CR) algorithm is proposed. The crossover recombination in DE is borrowed to enhance the local search and exploitation ability of original GBSO.

(3) The UAV path planning problem is solved by GBSO-CR. Numerical simulations have illustrated that GBSO-CR is competitive compared with the GBSO and other compared algorithms.

The rest of this paper is organized as follows. Sections 2 give a brief introduction of the mathematical problem formulation and constraint handling method. In Section 3, the GBSO-CR algorithm for UAV path planning is presented. The experimental results and discussion are presented in Section 4. Finally, the relevant conclusions and directions for future work are introduced in Section 5.

2. COST FUNCTION

In this paper, the point mass model of UAV is used for path planning. The main objective of UAV path planning is to search a safe, economic and flyable path to the destination with a minimal cost. This problem can be described as:

$$p_s(x_s, y_s, h_s) \xrightarrow{r_i} p_f(x_f, y_f, h_f) \tag{1}$$

Subject to:

$$\begin{cases} (x_l, y_l, z_l) \le r_i(x_i, y_i, z_i) \le (x_u, y_u, z_u) \\ c(r_i(t)) \le c_{\max} \\ |\theta_i| \le \theta_{\max} \\ r_i(t) \cap \square = \varnothing \end{cases}$$

p_s and p_f are the start and target points of the specific task, and r_i is the i-th segment of a specific path; c and θ are the spatial curvature and climb (or dive) angle; $\square$ denotes the obstacles and no-fly areas; (x_l, y_l, z_l) and (x_u, y_u, z_u) are the lower and upper boundaries of the search area.

The cost function is formulated as the sum of the threat and path length cost. The safety of a path is the major priority of UAV path planning. Furthermore, the flyability of a path is also considered as a constraint to 3D path planning. It contains the terrain, the maximal curvature and maximal climb (or dive) angle. Therefore, the UAV path planning problem can be formulated as a constrained single objective optimization problem with the following cost function:

$$\text{Min}: f_{cost} = \sum_{i=1}^{2} C_{safe}^{i} + C_{economic} \tag{2}$$

Subject to:

$$\sum_{i=1}^{3} G_{flyably}^{i} \le 0$$

The UAV path is represented by the cubic B-spine curve, whose characteristic polygon vertices are taken as the optimization variables. For a specific path R (r_i) (i =1,2,…N), r_i(t) is the i-th segment of R. The r_i(t) is generated by a predetermined number of three-dimensional points (x_i, y_i, z_i) using cubic B-spline curves. Then, the f_{cost} of R can be obtained by equation (2). The purpose of path optimization is to find the path with the minimum cost function while satisfying the constraints. C^1_{safe} and C^2_{safe} are the functions to prevent the UAV from being detected and destroyed, respectively. $C_{economic}$ is the length of UAV path to find a path with a shorter length and lower altitude by reducing the energy consumption. The $G^1_{flyably}$, $G^2_{flyably}$ and $G^3_{flyably}$ are constraints of the UAV path planning. $G^1_{flyably}$ is the function to prevent the i-th segment path r_i(t) of the path from collision with the terrain, $G^2_{flyably}$ and $G^3_{flyably}$ are the functions ensuring a path to satisfy the kinematic and dynamic constraints for UAV flight.

2.1. Safety

In the hostile environment, the safety is one primary objective for UAV path planning. This paper adopts a simple version of the radar and artillery model to assess the safety factor for UAV path planning [18].

2.1.1. Minimal risk of radar detection

The mathematical model of radar as follows:

$$C_{safe}^{1} = \sum_{i=1}^{N} f_i^1 \tag{3}$$

where

$$f_i^1 = \begin{cases} \delta_k \times \sqrt{(d_{PRD}^k)^2 - (d_{i,k})^2} & \text{if } d_{i,k} \le d_{PRD}^k \\ 0 & \text{otherwise} \end{cases}$$

f_i^1 is the cost function of radar detection for the i-th segment of a path, δ_k and d^k_{PRD} are the inherent parameters of the k-th radar, and $d_{i,k}$ is the distance between the i-th segment of a path and the k-th radar. The UAV closer to the radar, the higher probability of the UAV will be detected.

2.1.2. Minimal risk of destroyed

The mathematical model of artillery as follows:

$$C_{safe}^2 = \sum_{i=1}^{N} f_i^2 , \tag{4}$$

where

$$f_i^2 = \begin{cases} R_m \times \sqrt{(d_{PK}^m)^2 - (d_{i,m})^2} & \text{if } d_{i,m} \le d_{PK}^m \\ 0 & \text{otherwise} \end{cases} .$$

f_i^2 is the cost function associated with the risk of the *i*-th segment of a path, R_m and d^m_{PK} are the inherent parameters of the *m*-th missile, and $d_{i,m}$ is the distance between the *i*-th segment of a path and the *m*-th missile.

2.2. Economy

A path with a shorter length and lower altitude is more economic than that with a longer length and higher altitude.

$$C_{economic} = \sum_{i=1}^{N} f_i^3 , \tag{5}$$

where

$$f_i^3 = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2}$$

f_i^3 is the cost function associated with the length of the *i*-th segment of a path.

2.3. Flyable

If a path satisfies all the constraints simultaneously, it is regarded as a flyable path. Otherwise, it is an unflyable path.

2.3.1. Terrain constraint

An approximate cell decomposition of the terrain is adopted to convert the 3D space into a 2D matrix, which is named the map matrix [19]. It can discrete the continuous terrain dividing the 3D space into a grid. The terrain belongs to same grid has the same terrain height.

$$G_{flyable}^1 = \sum_{i=1}^{N} g_i^1 , \tag{6}$$

where

$$g_i^1 = \begin{cases} 0 & \text{if } z_i \ge (H_{map}(x_i, y_i) + h_{\min}) \\ z_i - (H_{map}(x_i, y_i) + h_{\min}) & \text{otherwise} \end{cases} .$$

g_i^1 represents the terrain constraint for the *i*-th segment of a path. $H_{map}(x_i,y_i)$ is the terrain height at point (x_i,y_i) obtained from the terrain map matrix, and $h_{\min}$ is the minimum safety value of the flight path above the terrain.

2.3.2. Maximal curvature constraint

$$G_{flyable}^2 = \sum_{i=1}^{N} g_i^2 , \tag{7}$$

where

$$g_i^2 = \begin{cases} 0 & \text{if } c_i \le c_{\max} \\ c_i - c_{\max} & \text{otherwise} \end{cases} .$$

g_i^2 is the curvature constraint for the *i*-th segment of a path, c_i and $c_{\max}$ are the approximate curvature for the *i*-th segment of a path and the maximum curvature constraint for UAV flight, respectively.

2.3.3. Maximal climb (or dive) angle constraint

$$G_{flyable}^3 = \sum_{i=1}^{N} g_i^3, \tag{8}$$

where

$$g_i^3 = \begin{cases} 0 & \text{if } |\theta_i| \le \theta_{\max} \\ |\theta_i| - \theta_{\max} & \text{otherwise} \end{cases}.$$

g_i^3 is the climb (or dive) angle constraint for the i-th segment of a path, and θ_{max} is the maximum climb(or dive) angle constraint of a path for UAV flight. The constraints are handled by the improved ε level comparison.

2.4. Constraint Handling

In this paper, the path planning is formulated as a complex optimization problem with terrain, maximum curvature and maximum climb (or dive) angle constraints. The methods of constraint handling as follow [20]. For a specific ε, an individual in_a is considered better than in_b according to the equation (9).

$$in_a <_\varepsilon in_b = \begin{cases} f(in_a) < f(in_b) & \text{if } \lambda(in_a), \lambda(in_b) \le 0 \\ f(in_a) < f(in_b) & \text{if } \lambda(in_a) = \lambda(in_b) \\ \lambda(in_a) < \lambda(in_b) & \text{otherwise} \end{cases} \tag{9}$$

f and λ are the objective function and the total overall constraint violation value, respectively. The update rules of ε are as equation (10).

$$\varepsilon(gen) = \begin{cases} \alpha_\rho & gen = 0 \\ \varepsilon(gen-1)(1 - FEs/T_c)^{cp} & \text{if } r_k < \beta \text{ and } FEs < T_c \\ (1+\tau)\phi_{\max} & \text{if } r_k \ge \beta \text{ and } FEs < T_c \\ 0 & \text{if } FEs \ge T_c \end{cases}. \tag{10}$$

$\varepsilon(gen)$ is the value of ε and r_k is the proportion of feasible solutions in the generation gen; α_ρ is the top ρ-th overall constraint violation of all individuals in the initial population; T_c is the control parameter. When the number of function evaluations (FEs) reaches T_c, the value of ε will be set as 0. Note that the ρ is set 25, β is set 0.8 and other parameters are set as suggested value in [20]. Thus, individuals can be evaluated by the cost function and constraint handling method.

3. GBSO-CR FOR UAV PATH PLANNING

Since the NP-hardness of UAV path planning, it is difficult to obtain the optimal solution using conventional mathematical optimization techniques. Therefore, we propose a novel GBSO-CR evolutionary framework (Table 1). For 3D path planning in threat environment, the more powerful local search and exploitation ability of algorithm the better for finding a more safe and economic solution. In order to solve the UAV path planning problem effectively, the crossover recombination is borrowed in GBSO-CR. The crossover recombination (Table 2) has the great potential to solve path planning problem, as it can interchange the infeasible waypoints with the feasible waypoints of other individual to avoid obstacles and risks. Further, the characteristic of local adjustment in the B-spline curve is also compatible with the purpose of using crossover recombination for path planning. The details of proposed GBSO-CR for UAV path planning are as follows: Firstly, randomly initialize NP individuals (x_i, y_i) within (x_l, y_l) and (x_u, y_u). Based on the environment and map modeling, the terrain heights of discrete position points are extracted from the map matrix [19]. The individuals are initially generated based on the terrain height, then initial height (z_i) of a path can be obtained, which must be higher than the terrain height. The $r_i(t)$ is generated by a predetermined number of three-dimensional points (x_i, y_i, z_i) using cubic B-spline curves. Secondly, the NP individuals are sent to the iteration circulation of GBSO-CR algorithm. Next, new individuals are generated by proposed algorithm for finding better individuals. Finally, the best individual will be selected according to the cost function of each individual (equations (2–8)) and the ε level comparison (equation (9)).

Table 1

Procedure of the GBSO-CR algorithm

Input: NP, m, $p_{onecluster}$, $p_{onecenter}$, $p_{twocenters}$, CR, (x_l, y_l, z_l) and (x_u, y_u, z_u)
Output: the best individual

Randomly initialize NP individuals;
Generate UAV paths using the characteristic polygon vertices of B-spline curve in NP individuals;
Evaluate NP paths (equations (2–8))and find the best individual *GlobalBest*;
While (*Currentiter*⩽*Maxiter*) do
  Cluster NP into m groups by the cost function f_{cost} and equation (9);
  For idx =1 to NP do
    for j =1 to D do
      If $rand < p_{onecluster}$ then
        Randomly select a subgroup $group\{gr\}$;
          if $rand < p_{onecenter}$
            $Donor\_idx(j) = center(gr, j)$;
          else
            Randomly select an individual k in $group\{gr\}$;
            $Donor\_idx(j) = group\{gr\}(k, j)$;
          end if
      Else
          Randomly select two clusters $r1$ and $r2$ ($r1 \neq r2$);
          Randomly select two individual $r3$ and $r4$ ($r3 \neq r4$);
          $r = \text{rand}$;
          if $rand < p_{twocenter}$
            $Donor\_idx(j) = r \times center(r1, j) + (1-r) \times center(r2, j)$;
          else
            $Donor\_idx(j) = group\{r1\}(r3, j) + (1-r) \times group\{r2\}(r4, j)$;
          Endif
        end**If**
    endfor
  ENDFor
  Get *Trial_idx* use crossover recombination (Table 2);
  $\xi = \text{rand} \times e^{1-\frac{MaxIterations}{MaxIterations-CurrentIteration+1}} \times step$
  $New\_idx = Trial\_idx + \xi \times N(\mu, 0)$;
  Generate UAV paths using the characteristic polygon vertices of B-spline curves in *New_idx*;
  Evaluate the new path by equations (2–8) and equation (9);
  Update NP individuals and *GlobalBest*;
  Update ε using equation (10);
ENDWhile
Return the best individual

Table 2

Procedure of crossover recombination

Input: D, CR, *Donor_idx* and *GlobalBest*
Output: *Trial_idx*

$G\_index = \text{rand}(1, D) < CR$; (A< B, returns the logical operation value of whether 'A' is less than 'B')
$D\_index = G\_index < 0.5$;
$Trial\_idx = Donor\_idx \odot D\_index + GlobalBest \odot G\_index$ ("$\odot$" is Hadamard product).

4. SIMULATION RESULTS AND DISCUSSIONS

In this section, we apply our proposed GBSO-CR to solve the UAV path planning in complex environment. The prototype system of UAV path planning was implemented with MATLAB (R2021a) based on the proposed method. To prove the effectiveness of the proposed, comparison analysis was conducted for the five algorithms using the platform "PlatEMO" [21], i.e., GA, DE, Success-history based parameter adaptation for differential evolution algorithm (SHADE), GBSO, and GBSO-CR. The download link of the programs and example can be obtained from the URL: https://www.researchgate.net/project/GBSO-CR-for-path-planning.

The mission area of the UAV was 1000 km long, 600 km wide and 0.5 km high with the given terrain and threat areas. The threat areas were indicated by circles in 2D and cylinders in 3D. The number of waypoints was set to 13, then the number of optimization variables was 39 (3D points). The number of optimization variables was set by the user. For finding a flyable path for UAV, the maximum curvature c_{max} was set as 1/30, and the maximum climb (or dive) angle θ_{max} was 30 degree. The c_{max} and the θ_{max} depend on the performance of UAV.

The purpose of the simulation scenario is to test whether an algorithm can find a flyable UAV path from starting point to the target point with economic and lower risk. The detailed parameters of the simulation scenario are listed in Table 3 [22]. Table 4 shows the parameter settings of each algorithm for the simulation scenario. Those values were set according to the recommended value of each algorithm. The crossover rate (*CR*) denotes the contribution rate of global-best individuals to the generation of new individuals and the recommended value of *CR* is [0, 0.2] [23]. Then 0.1 was adopted in the proposed GBSO-CR algorithm for balancing the search capability and the convergence speed. For comparison analysis, 25 independent runs were executed for each algorithm.

Table 3

Parameters of the scenario

Category	Location and threat radius (km)
Rader 1	[(260, 370), 80]
Rader 2	[(450, 90), 80]
Rader 3	[(800, 230), 100]
Artillery 1	[(305, 196), 100]
Artillery 2	[(800, 400), 80]
No-fly zone 1	[(350, 500), 100]
No-fly zone 2	[(600, 300), 60]

Table 4

Algorithm parameters for comparison

Algorithm	Related parameters
GA	NP =100, Pc =10, $disC$ =20, $proM$ =1, $disM$ =1
DE	NP =25, CR =1, F=0.5
SHADE	NP =100
GBSO	NP =25, m =5, $p_{onecluster}$ = 0.8, $p_{onecenter}$ = 0.4, $p_{twocenters}$ = 0.5
GBSO-CR	NP =25, m =5, CR= 0.1, $p_{onecluster}$ = 0.8, $p_{onecenter}$ = 0.4, $p_{twocenters}$ = 0.5
Stopping criterion	the number of function evaluations is 20000

The cost function, runtime and feasible rate (*f.rate*) of the results were listed to verify the effectiveness of the proposed algorithm. The feasible rate is defined by

$$f \cdot rate = number\ of\ \ feasible\ solutions\ /\ NP \tag{11}$$

number of feasible solutions denotes the number of solutions that meet the terrain constraint, the spatial curvature and climb (or dive) angle constraints for UAV flight at the end of each run, and *NP* is the number of population. The *f.rate* can measure the number of flyable paths found by the algorithm. The higher value of feasible rate, the better performance of the algorithm is. To ensure a statistically sound conclusion, a

Wilcoxon rank-sum test with a 0.05 significance level was used to show the statistically significant differences in the performance results. Note that the symbols "+," "–,"and "~" respectively, indicate the compared algorithms perform significantly better, worse, and similarly when compared with GBSO-CR for solving the path planning problems.

There are safe spaces between the "Artillery 1" and "Radar 1", "Artillery 1" and "Radar 2" from the starting point to the target point. The optimal UAV paths in 2D and 3D obtained by the five algorithms are shown in Fig. 1 and Fig.2.

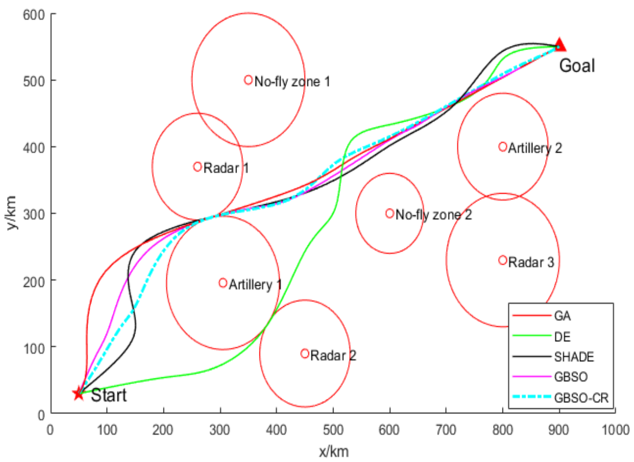

Fig. 1 – The 2D paths obtained by the five algorithms.

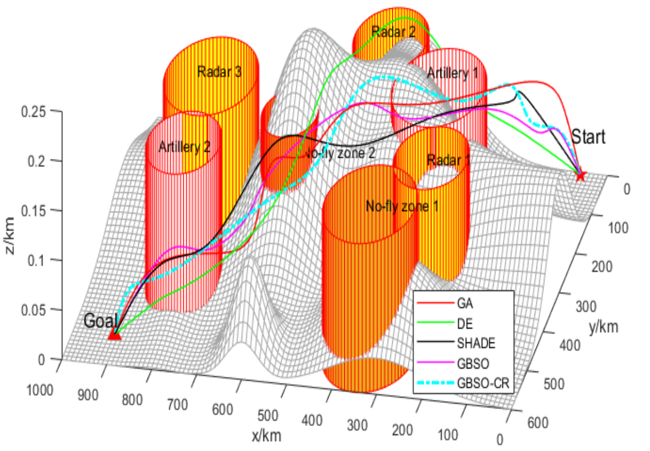

Fig. 2 – The 3D paths obtained by the five algorithms.

As shown in figures, the paths are gathered in the reserved safe space. All the algorithms can find feasible paths that meet the requirements of the terrain, spatial curvature and climb (or dive) angle constraints for UAV. The statistical results by the five algorithms during 25 runs are presented in Table 5.

Table 5

The f_{cost}, runtime and feasible rate of the five algorithms

	Algorithm	GA	DE	SHADE	GBSO	GBSO-CR
	mean	1093.5	1409.2	1087.6	1070.4	**1065.3**
f_{cost}	std	25.4	439.0	77.6	8.70	**3.85**
	Rank sum test	-	-	-	-	
	mean	**11.28**	11.53	11.29	15.28	13.06
runtime(s)	std	0.70	**0.58**	0.59	1.08	0.62
	Rank sum test	+	+	+	~	
	mean	**1.000**	0.985	**1.000**	0.982	0.995
f_{rate}	std	**0**	0.034	**0**	0.020	0.017
	Rank sum test	~	~	~	-	

As shown in Table 5, several observations can be obtained:

(i) The ascending sequence in terms of f_{cost} is GBSO-CR < GBSO < GA < SHADE < DE, the GBSO-CR had the smallest value 1065.3. It demonstrating that the GBSO-CR has capable of finding the best path with short length and low risk among compared algorithms. The result of GBSO-CR (1065.3) better than GBSO (1070.4) showing that the crossover recombination is capable of improving search ability of GBSO. Moreover, the standard deviation of GBSO-CR which is 3.85 is also smaller than those of the other algorithms. This further demonstrates the high robustness of GBSO-CR.

(ii) In terms of runtime, the GBSO and GBSO-CR have worse results, since the per-variable updates generate one problem variable at a time rather than all variables in one step. The per-variable update has the nature advantage of improving local search ability, but it brings longer running time. With the development of computer technology, the runtime of algorithm will be shorter.

(iii) It can be seen that the best result in terms of *f.rate* is achieved by GA and SHADE. However, the f_{cost} of GA and SHADE are extremely worse than that of GBSO-CR. The value of GBSO-CR (0.995) better than GBSO (0.982) demonstrating the GBSO-CR has higher search ability. In other hand, the results demonstrating that the ε level comparison is effective to solve this constrained 3D UAV path planning optimization problem.

Additionally, in order to perform convergence analysis of the GBSO-CR, the values of f_{cost} by the five algorithms are shown in Fig. 3. The *y*-axis of cost curve is the best individual found when corresponding to the *x*-axis number of evaluations. As shown Fig. 3, the value of the cost function may increase with the number of iterations instead of decreasing all the time. For a specific ε, an individual with larger f_{cost} and small constraint value is better than that with smaller f_{cost} and a larger constraint value.

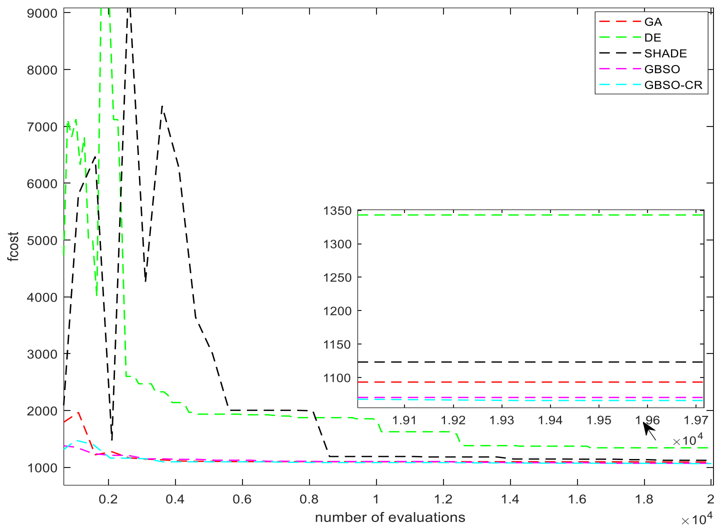

Fig. 3 – The f_{cost} obtained by the five algorithms.

For this comparison, the GBSO-CR has a fast convergence rate, and it can find a better result using a smaller number of evaluations. Combined Fig. 3 with Table 5, it shows that GBSO-CR is superior to those of the other four algorithms in terms of convergence rate, the lowest average and standard deviation value of its solutions.

5. CONCLUSIONS

This paper presents a new GBSO-CR algorithm for constrained 3D UAV path planning. This GBSO-CR combines crossover recombination with GBSO algorithm for improving search capability and robustness of GBSO. The cost function is designed according to the safety, economy and flyability of 3D UAV path. The cubic B-spline curve is used to represent UAV paths to ensure the path smoothness. Simulation results and comparison analysis demonstrate that the proposed GBSO-CR is significantly superior to GBSO and compared algorithms for UAV path planning. In future research work, the proposed algorithm will be combined with advanced artificial intelligence techniques, such as machine learning, neural network and fuzzy optimization, and to study the path planning problem for multiple UAVs with multi-objectives optimization.

ACKNOWLEDGMENTS

We would like to thank reviewers and editors for useful suggestions. This work was partially supported by the project of the National Natural Science Foundation of China (Grant No 41904028), the Shaanxi Province Key Research and Development Projects (Grant No 2018ZDXM-GY-024), and the Science, Technology and Innovation Commission of Shenzhen Municipality (Grant No JCYJ20180306171439979).

REFERENCES

1. E.T. ALOTAIBI, S.S. ALQEFARI, A. KOUBAA, *LSAR: Multi-UAV collaboration for search and rescue missions*, IEEE Access, **7**, pp. 55817–55832, 2019.
2. M. ATIF, R. AHMAD, W. AHMAD, L. ZHAO, J. J.P.C. RODRIGUES, *UAV-assisted wireless localization for search and rescue*, IEEE Systems Journal, **15**, *3*, pp. 3261–3272, 2021.
3. C. ZHOU, R.H. XIONG, H.Z. ZENG, J. XIAO, Y. WANG, P.F. JIA, J. YE, T.T. Zhao, K. HU, *Aerial locating method design for civil aviation RFI: UAV monitoring platform and ground terminal system*, Journal of Intelligent & Robotic Systems, **103**, *2*, pp. 1–13, 2021.
4. E. BESADA-PORTAS, L. DE LA TORRE, J.M. DE LA CRUZ, B.D. ANDRÉS-TORO, *Evolutionary trajectory planner for multiple UAVs in realistic scenarios*, IEEE Transactions on Robotics, **26**, *4*, pp. 619–634, 2010.
5. Y.B. CHEN, J.Q. YU, Y.S. MEI, Y.F. WANG, X.L. SU, *Modified central force optimization* (*MCFO*) *algorithm for 3D UAV path planning*, Neurocomputing, **171**, pp. 878–888, 2016.
6. Y.G. FU, M.Y. DING, C.P. ZHOU, H.P. HU, *Route planning for unmanned aerial vehicle* (*UAV*) *on the sea using hybrid differential evolution and quantum behaved particle swarm optimization*, IEEE Transactions on Systems, Man, and Cybernetics: Systems, **43**, *6*, pp. 1451–1465, 2013.
7. X.B. YU, C.L. LI, J.F. ZHOU, *A constrained differential evolution algorithm to solve UAV path planning in disaster scenarios*, Knowledge-Based Systems, **204**, pp. 1–11, 2020.
8. J. MANDAL, A. GOSWAMI, J.W. WANG, M.K. TIWARI, *Optimization of vehicle speed for batches to minimize supply chain cost under uncertain demand*, Information Sciences, **515**, pp. 26–43, 2020.
9. E. OSABA, X.S. YANG, I. FISTER, J.D. SER, P. LOPEZ-GARCIA, A.J. VAZQUEZ-PARDAVILA, *A discrete and improved bat algorithm for solving a medical goods distribution problem with pharmacological waste collection*, Swarm and Evolutionary Computation, **44**, pp. 273–286, 2019.
10. B. BELDJILALI, B. BENADDA, Z. SADOUNI, *Vehicles circuits optimization by combining GPS/GSM information with metaheuristic algorithms*, Romanian Journal of Information Science and Technology, **23**, pp. 5–17, 2020.
11. R.J. SZCZERBA, *Threat netting for real-time, intelligent route planners*, Proceedings of the 1999 Information, Decision and Control. Data and Information Fusion Symposium, Signal Processing and Communications Symposium and Decision and Control Symposium, 1999, pp. 377–382.
12. J.M.O. KANE, D.A. SHELL, *Concise planning and filtering: hardness and algorithms*, IEEE Transactions on Automation Science and Engineering, **14**, *4*, pp. 1666–1681, 2017.
13. S. CHENG, Q.D. QIN, J.F. CHEN, Y.H. SHI, *Brain storm optimization algorithm*: *a review*, Artificial Intelligence Review, **46**, *4*, pp.445–458, 2016.
14. H.X. QIU, H.B. DUAN, *Receding horizon control for multiple UAV formation flight based on modified brain storm optimization*, Nonlinear dynamics, **78**, *3*, pp. 1973–1988, 2014.
15. Y. YU, S.C. GAO, Y.R. WANG, J.J. CHENG, T. YUKI, *ASBSO*: *An improved brain storm optimization with flexible search length and memory-based selection*, IEEE Access, **6**, pp. 36977–36994, 2018.
16. M. EI-ABD, *Global-best brain storm optimization algorithm*, Swarm and evolutionary computation, **37**, pp. 27–44, 2017.
17. R. STOM, K. PRICE, *Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces*, Journal of Global Optimization, **11**, pp. 341–359, 1997.
18. S. LEE, Y. KIM, *Optimal output trajectory shaping using Bézier curves*, Journal of Guidance, Control, and Dynamics, **44**, *5*, pp. 1027–1035, 2021.
19. V. ROBERGE, M. TARBOUCHI, G. LABONTE, *Comparison of parallel genetic algorithm and particle swarm optimization for real-time uav path planning*, IEEE Transactions on Industrial Informatics, **9**, *1*, pp. 132–141, 2013.
20. Z. FAN, Y. FANG, W.J. LI, Y.T. YUAN, Z.J. WANG, X.C. BIAN, *LSHADE44 with an improved ϵ constraint-handling method for solving constrained single-objective optimization problems*, Proceedings of the IEEE Congress on Evolutionary Computation, 2018, pp. 1–8.
21. Y. TIAN, R. CHENG, X.Y. ZHANG, Y.C. JIN, *PlatEMO*: *A Matlab platform for evolutionary multi-objective optimization* [educational forum], IEEE Computational Intelligence Magazine, **12**, *4*, pp. 73–87, 2017.
22. X.Y. ZHANG, H.B. DUAN, *An improved constrained differential evolution algorithm for unmanned aerial vehicle global route planning*, Applied Soft Computing, **26**, pp. 270–284, 2015.
23. J. RONKKONEN, S. KUKKONEN, K.V. PRICE, *Real parameter optimization with differential evolution*, Proceedings of the 2005 IEEE Congress on Evolutionary Computation, **1**, pp. 506–513, 2005.

Received November 30, 2021