

# Improved Black Hole Algorithm for Efficient Low Observable UCAV Path Planning in Constrained Aerospace

A. A. Heidari\*, R. A. Abbaspour

Department of Surveying and Geomatics Engineering, College of Engineering, University of Tehran  
Tehran, Iran

as\_heidari@ut.ac.ir  
abaspour@ut.ac.ir

## Abstract

An essential task of UAV autonomy is automatic path planning. There are many evolutionary planners for Unmanned Aerial Vehicles (UAVs) that have been developed UAV community. In this paper a comparative study about performance of effective trajectory planners is done. Also an efficient version of black hole methodology has been introduced for single UCAV trajectory planning, and an enhancement is designed to communicate among stars and black hole based on relativity theory principles. By considering UCAV Dynamic properties and environment constraints, Developed path planner based on black hole algorithm can compute feasible and quasi-optimal trajectories for UCAV flight. Our comparison of algorithms shows that IBH generates desired optimal trajectories. Then path planning task of UCAV is performed. Simulations show advantage of IBH methodology.

**Keywords:** *Unmanned combat aerial vehicle (UCAV), Flight Simulation, Trajectory Planning, black hole algorithm*

## 1. Introduction

There are many evolutionary planners for Unmanned Aerial Vehicles (UAVs) for optimal Navigation in UAV community. UCAV is from the family of unmanned aircrafts developed for performing reconnaissance missions. Long-range drones have an autopilot system for following predesigned way-points and continue motion based on planned mission, when they are out of the control of station's communication range. Operational drones need human control, but operator tasks are based on UCAV level of autonomy.

Many activities could be done to UCAV systems to reach to autonomous navigation. These steps maybe include mapping environment, onboard DTM generation, trajectory planning, and control systems. Path planning is a complex problem in the autonomous navigation. Its objective is to find an optimal constrained flight path in proper time to UAV be able to accomplish mission tasks. Choosing efficient algorithms for solving path planning

problem is an influential step. Optimal path planning relies on optimization technics so it's usually solved offline.

Use of UCAVs, which can fly autonomously in aerospace environments, is necessary. Reliable safe navigation of UCAV in Complex missions has technical challenges and UCAV planning is an essential task. Aerospace applications of UCAVs require exact maneuvers and optimal decisions and robust path planning algorithms. Complex space around UCAV flight trajectory makes the problem NP-hard.

Making UCAVs more autonomous for performing automated take-off and landing, target recognition and path planning, is a vital task in future aeronautics. Path planning is designing a chain of events such that an object can move in order to reposition from a beginning situation to a goal position. Path planning is vitally in search, surveillance, and tracking missions. Planning algorithm is a series of steps to compute plan by enough cognizance of environment and some constraints. The planned UCAV trajectory should avoid the obstructions and satisfy the UCAV's mission requirements. Any constraint is related to UCAV model and environment.

On the other hand, many evolutionary approaches based on natural concepts have been proposed [1]. So many Evolutionary Algorithms (EA) [2], have been developed for solving this problems. The most well-known evolutionary method, genetic algorithms (GAs) is an adaptive strategy and based on Darwin's natural selection theorem [3]. Particle swarm optimization (PSO) is another technique that is inspired from the social behavior of birds [4]. Ant colony optimization (ACO) is a cooperative search technique that is motivated from ant colonies [5]. Some of the other well-known heuristic approaches are Simulated Annealing (SA) [6], Tabu Search (TS) [7], Honey Bee Mating Optimization (HBMO) [8], Modify Imperialist Competitive Algorithm (MICA) [9] and artificial bee colony (ABC) [10].

Based on pervious works, path planning problem was presented to new hybrid techniques based on neural network [11], fuzzy logic [12], ACO [13], PSO [14, 15],

GA [16] and the artificial potential field [17]. When we have large mission ranges in UCAV flight, trajectory planning will be a large scale constrained optimization process. General methods on 3D path planning could be applied to solve this NP-hard problem including graph search like A\* [18] and D\* and rapidly exploring Random Trees (RRT) [19] and other is potential fields, evolutionary techniques include PSO, GA, ACO and multi-objective evolutionary algorithms [20]. Every method has its own robustness in certain aspects that is related to the problem complexity. By using Evolutionary based approaches for enhancing UCAV operational autonomy, we can combine flight dynamics, physical constraints, and mission objectives in the form of mathematical model.

The structure of the paper is as follows. In Section 2 black hole algorithm is introduced and improved black hole algorithm is proposed .in Section 3 defines the UCAV planning problem and section 4 holds the main results of UCAV simulation in 2D environment. Conclusion is the last section.

## 2. Principle of Black Hole Algorithm

The concept of a black hole is developed based on Einstein's general theory of relativity that explained as infinite curvature of space-time. Every nearby object can't escape from this gravitational field, including light. Firstly, idea of such a mass concentration proposed by Laplace in the 18th century. Cosmologists proved that massive stars with no fuel will collapse into black holes and make some strange distortion in space. In particular, a black hole has some sphere-shaped boundary known as the 'event horizon.' Inside of the horizon, it's impossible to escape from singularity of black hole. The radius of such a horizon is named as Schwarzschild radius that is defined as Eq. (1):

$$R = \frac{2GM}{c^2} \quad (1)$$

Where G is the gravitational constant, M shows the mass of black hole, and c is speed of light. Back in the 1970's, Stephen Hawking proposed that black holes produce some radiations that come from their enormous mass.

The black hole algorithm (BH) firstly proposed by Hatamlou [21]. Similar to other swarm based methods, a population of candidate solutions according to a given problem is generated randomly in the search space. The population-based methodologies evolve the current population to find optimal solutions via certain procedures. BH evolves the population by assimilating all the stars toward the black hole, and replacing those stars that passed through the horizon by newly generated candidates. Black holes are real candidates of the population. Then, all the candidates are moved to the black hole. This operation is

based on their Current location and a random number. The details of the BH algorithms are presented as follows:

After initializing step, fitness values of stars is computed and the best candidate in the population should be black hole and the rest form the normal stars. The black hole can pull the stars that are in the neighborhood. After initializing black hole and stars, the black hole starts absorbing the stars around it and all the stars start moving towards the black hole. The absorption of stars by the black hole is formulated as Eq. (2):

$$x_i(t+1) = x_i(t) + rand \times (x_{BH} - x_i(t)) \quad i=1,2,3,...,n \quad (2)$$

Where  $x_i(t)$  and  $x_i(t+1)$  donate the locations of the  $i$ th star at iterations t and t+1, respectively.  $x_{BH}$  shows the location of black hole in the search space. Rand is a random number in the interval [0, 1] and N is the number of stars. After moving towards the best candidate, if the cost of a star was lower than the black hole, the position should be exchanged. During this process, for each star, there is a probability of crossing of the event horizon. When a candidate vanished, another star is born randomly in the search space. The radius of event horizon is computed based on the Eq. (3):

$$R = f_{BH} / \sum_{i=1}^N f_i \quad (3)$$

Where  $f_{BH}$  donates the fitness value of black hole,  $f_i$  is fitness of the  $i$ th star and N is the number of stars (candidate solutions).

## 3. Improved Black Hole Algorithm

Black hole algorithm is weak to perform global search perfectly in the big problem spaces. We should improve the absorption process in black hole algorithm. Two important features of the swarm-based methods are exploration and exploitation. The exploration is related to searching of space, where the exploitation is hunting the optimum [22]. The exploration is a significant theme in swarm-based heuristic algorithms. Over time, exploring will be reduced and exploitation ability fades in, so the algorithm adjusts itself in the semi-optimal points. There should be a balance between exploration and exploitation, to keep black hole algorithm safe from trapping in local optima.

In our work, black hole algorithm will be improved, using stars gravities information (see figure 1). For this aim, kind of gravitational force between stars is defined and the movement of stars to the black hole is adjusted during the searching solution space.

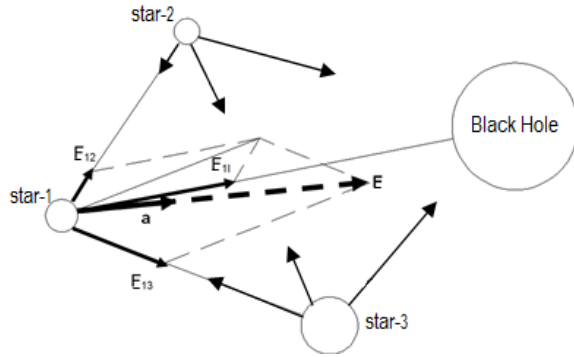


Fig. 1 Proposed movement process

We have a swarm with  $N$  stars. The position of the  $i^{\text{th}}$  stars ( $X_i$ ) is defined by Eq. (4).

$$X_i = (star_i, \dots, star_N, blackhole_d) \quad (4)$$

Where  $star_i$  is the position of  $i^{\text{th}}$  star and  $blackhole_d$  is the position of  $d^{\text{th}}$  black hole, respectively. At a specific time 't', we define the absorption acting on star 'i' from star 'j' as Eq. (5).

$$E_{ij}^d(t) = \zeta(t_0) \frac{C_{pi}(t) \times C_{aj}(t) \times (star_j(t) - star_i(t))}{(D_{ij}(t) + \epsilon)^2 \times (C_{pi}(t) + C_{aj}(t))} \times \left(\frac{t_0}{t - t_0}\right)^\alpha \quad (5)$$

Where  $C_{aj}$  is the power of star  $j$ ,  $C_{pi}$  is the power related to star  $i$ ,  $\zeta(t_0)$  is initial absorption constant,  $\epsilon$  is a small constant, and  $D_{ij}(t)$  is distance between two stars  $i$  and  $j$ .

To give a stochastic characteristic to black hole algorithm, total force is randomly weighted sum of the forces of others (see Eq. (6)).

$$E_i^d(t) = \sum_{j=1, j \neq i}^N rand_j E_{ij}^d(t) \quad (6)$$

Where  $rand_j$  is in  $[0,1]$ . Hence, the acceleration of the star  $i$  at time  $t$ , and in direction  $d^{\text{th}}$ , is given by Eq. (7).

$$a_i^d(t) = \frac{E_i^d(t)}{C_{ii}(t)} \quad (7)$$

Where  $C_{ii}$  is the Power of  $i^{\text{th}}$  star, the next velocity of star is considered as follows. Therefore, position and its velocity are calculated based on Eq. (8) and Eq. (9).

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \quad (8)$$

$$star_i(t+1) = star_i(t) + v_i^d(t+1) \quad (9)$$

Where  $rand_i$  is in  $[0, 1]$ . This random number is for randomization of the search.

#### 4. UCAV Path Planning Problem

Unmanned air systems should be capable to perform surveillance missions with considering a variety of

objectives [23]. There are several considerations for an efficient path planner including: optimality, completeness and complexity, which are related to vehicle motion dynamics. The extra dimensions of UCAV-PP problem increase computational complexity for the evolutionary planner, because the design space is extended. Also Planners should be able to solve constrained optimization problems. Modeling of threat sources in UCAV environment is the important task in path planning. There are some threatening areas include radars, artilleries and missiles that is modeled in the shape of circles. The probability of detection or crashing is proportional to inverse distance from the center of threats. We are seeking optimal paths in such a 2D environment (see figure. 2)

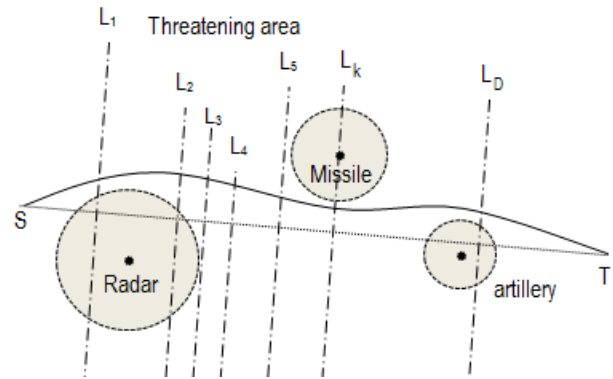


Fig.2 proposed model of environment

First we connect start point S to the end point T and then divide this line to the D equal line segment. Then we draw another lines perpendicular to ST, defined as  $L_1, L_2, L_3, \dots, L_k, \dots, L_{D-1}$ . By selecting a discrete point at every vertical line  $L_k$ , there are a collection of points  $G = \{S, L_1(x(1), y(1)), L_2(x(2), y(2)), \dots, L_k(x(k), y(k)), \dots, L_{D-1}(x(D-1), y(D-1)), T\}$ . Any trajectory is produced by connecting the point in order. We should optimize consequence of coordinates to obtain a better fitness of objective function. Performance evaluation of UAV trajectory planning is based on threat cost, fuel consumption, path length constraints, length of the path passage, and formula in Eq. (10):

$$\text{Min } H = \int_0^L [q_1 C_t + q_2 C_f + q_3 C_a + q_4 C_l] ds \quad (10)$$

Where  $H$  is the total cost,  $L$  is the length of produced path,  $C_t$  and  $C_f$  donates the threat cost and fuel cost of each line segment respectively,  $C_a$  shows biggest turning angle cost and  $C_l$  represent shortest path passage costs. We used 4 weighting coefficients that the values depend on the feasibility of related costs.

Table1: Comprehensive simulation results from UCAV path planners

Setting	Results	ICA	IBH	PSO	GA	ABC	MICA	HBMO	ACO	BH
Pop-Size	Best Normalized	5.7113	0.6930	3.6282	2.6478	3.6814	4.6638	2.3221	8.5262	1.4014
50	Worst Normalized	20.1727	5.4736	38.2519	11.9173	1.3392	24.1006	19.6268	17.9082	27.0725
Max-G	Mean Normalized	12.3252	3.0376	8.9141	5.2342	2.2745	17.2166	6.2579	15.2141	7.2934
50	Mean CPU time	1.2132	2.4991	2.0191	2.0079	1.7421	0.7786	1.6261	1.6723	2.7314
Pop-Size	Best Normalized	6.2729	0.6312	3.3957	2.5194	1.9831	4.9616	1.8520	11.5732	1.6971
50	Worst Normalized	13.8979	5.1789	24.8706	12.1778	17.4878	26.2727	13.6576	14.8904	26.7821
Max-G	Mean Normalized	7.2377	2.2128	7.1157	4.5231	8.5191	12.9574	4.1165	15.2780	5.4426
100	Mean CPU time	3.4431	5.1186	2.3429	3.1176	3.8980	2.4572	3.2482	5.5229	4.2529
Pop-Size	Best Normalized	4.8227	0.6817	3.4165	2.8952	1.7285	4.6888	0.7598	4.9981	1.6759
50	Worst Normalized	12.1476	5.2568	26.5682	16.2567	16.2542	38.2297	14.3342	19.4153	24.8980
Max-G	Mean Normalized	8.8585	1.1533	9.4109	3.1781	4.8751	10.5678	4.1580	15.1832	5.7809
150	Mean CPU time	4.4385	6.1378	3.8958	3.9539	3.2583	2.2530	3.6453	4.1830	6.3371
Pop-Size	Best Normalized	1.7810	1.2761	2.6759	1.2895	3.1588	2.8524	0.8187	10.2285	1.2931
50	Worst Normalized	8.4521	13.1872	27.5894	19.9254	12.3586	12.1581	5.2580	19.1589	24.5258
Max-G	Mean Normalized	1.5821	0.1584	7.1252	2.1581	3.1575	10.8456	3.2522	15.8984	2.4571
200	Mean CPU time	4.8970	5.1421	4.3558	4.9852	3.2524	2.7850	5.5852	4.9851	9.4589
Pop-Size	Best Normalized	1.0023	0.4548	3.4579	0.2581	0.4459	2.1589	0.4539	8.1257	1.1254
50	Worst Normalized	12.0125	4.4573	24.6581	9.4454	6.4572	19.6124	17.4582	11.0459	24.1245
Max-G	Mean Normalized	3.1259	1.1225	8.1243	3.1245	2.1259	10.1244	3.1249	17.0124	3.1244
250	Mean CPU time	5.1259	3.1244	4.1244	4.8127	6.1240	2.1242	6.1248	7.1245	12.1245

Computation of the costs is based on Eq. (11-12):

$$C_{angle} = \sum_{i=1}^n \left[ 1 - \frac{\tau_i^T \tau_{i+1}}{|\tau_i| |\tau_{i+1}|} / \cos \alpha \right] \quad (11)$$

$$C_{length} = \sum_{i=1}^k \frac{l_{\min} - l_i}{l_{\min}} \quad (12)$$

For computing threat cost, five points is selected along of each edge between two discrete points. If  $i$ -th edge is within the threat range, cost is calculated based on Eq. (13).

$$C_{t,L_{ij}} = \frac{L_{ij}^5}{5} \sum_{k=1}^{N_t} t_k \times \left( \frac{1}{d_{0.1,k}^4} + \frac{1}{d_{0.3,k}^4} + \frac{1}{d_{0.5,k}^4} + \frac{1}{d_{0.7,k}^4} + \frac{1}{d_{0.9,k}^4} \right) \quad (13)$$

Where  $N_t$  is the number threatening areas,  $L_{ij}$  is the trajectory between  $i$ -th and  $j$ -th points,  $d_{0.1,k}$  is the distance from the 1/10 point of  $L_{ij}$  to the  $k$ -th threat center and  $t_k$  is the threat level of  $k$ -th threat (see figure 3)

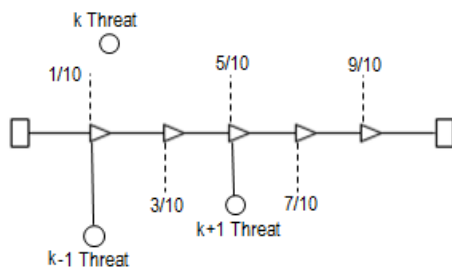


Fig.3. computation of threat cost

## 5. UCAV Flight Simulations and experiments

Simulations of proposed planners performed in the same computer and all the tests were under the same conditions. For performance analysis of the evolutionary planners, it was tested with different parameters. Each experiment was in loop to 100 times for reaching to reliable result.

We used MATLAB R2011b environment on a PC with 2.33 GHz Intel Core 2 Duo and 4 GB of RAM memory.

For performance evaluation of improved black hole algorithm, we compared that with other methods include GA, PSO, ACO, ABC, HBMO, ICA, MICA and BH. Each algorithm is tested by considering different numbers of control points, and threads. For performance analysis of algorithms, standard parameters selected that was same as [24]. For evaluating of Max-gen parameter effects, we performed 100 simulations for every algorithm to achieve reasonable results. Also, we subtract 60 from the actual values, i.e., value of 7.7113 is transformed number generated from the value 67.7113. The results of UCAV path planning simulations are presented in table 1.

Selecting best maximum generation value of algorithms is usually critical for related computational problems. The possibility of finding optimal solution is directly proportional with increasing of Max-gen value and final achievement will be good searching of space. Based on results on table 1, IBH provides better results than others methods. For example, in the case with Max-gen of 150, IBH in compare with PSO, GA, ACO, and ICA provides



results with less diversion. The results of ABC and HBMO show that they their performances are almost similar. By comparing the results with Max-gen of 200 and 250, it's concluded that feasibility of the ICA, IBH, ABC, HBMO and MICA is the similar but GA and ACO have weak performances. From Table 1, it's concluded that the preformation of IBH is superior in comparison with other methods, while ABC and HBMO should be ranked as second best among all 5 categories.

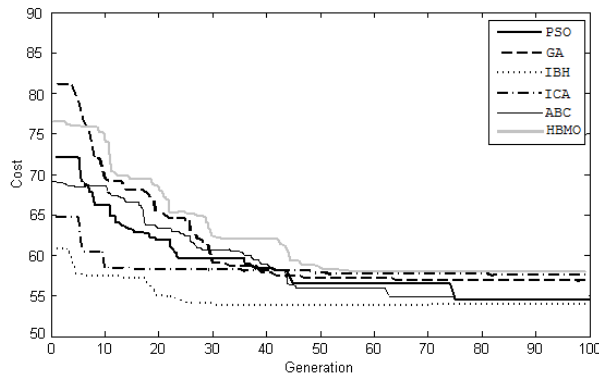


Fig.4. Comparison of conversion speed related to PSO, GA, IBH, ICA, ABC and HBMO for solving UCAV path planning problem

Based on Figure 4, IBH planner shows remarkable results both in optima solution quality and in convergence speed rather than other planners.

## 6. Conclusions

In this article, a new approach is proposed for trajectory planning in 2D constrained environment based on enhanced black hole algorithm. An efficient version of black hole methodology has been introduced for single UCAV trajectory computing, and an enhancement is designed to communicate among stars and black hole based on relativity theory principles. This approach will improve global convergence rate of black hole method while robustness of original algorithm is maintained. Then, decision maker can discover any safe path by connecting selected nodes, by considering threat sources and fuel consumptions.

This work can enhance the UCAV's offline optimal planning, navigation, and guidance in realistic missions. The proposed method based on IBH has superior performances in competition with other well-known methods. This study is part of system simulations on our previous work. Our method provides valid 2D safe path with low computational complexity; while control station can obtain sub-optimal routes based on mission requirements. The simulation results show that this novel algorithm not only can produce path with more robustness, but also has higher convergence speed than other implemented algorithms.

## References

- [1] Heidari, A. A, Meta-heuristic Algorithms for Optimization Problems Combined With Pseudo-code & Implementation in the Ruby Language, First Edition, Tehran: Mahvare; 2012.
- [2] H. Sarimveis and A. Nikolakopoulos, "A Line up Evolutionary Algorithm for Solving Nonlinear Constrained Optimization Problems", Computers & Operations Research, 32(6):pp.1499–1514, 2005.
- [3] Holland J. Adaptation in natural and artificial systems. Ann Arbor, MI: The University of Michigan Press; 1975.
- [4] Eberhart RC, Kennedy J. "A new optimizer using particle swarm theory". In: Proceedings of the 6th international symposium on micro machine and human science, 1995, Japan.
- [5] M. Dorigo, C. Blum, "Ant colony optimization theory: a survey", Theoretical Computer Science, 2005, pp.243–278.
- [6] D.S.Johnson, C.R.Aragon, L.A.McGeoch, C.Schevon, "Optimization by simulated annealing: experimental evaluation. Part I". Operations Research-37, 1989, pp.865–892.
- [7] M. Gendreau, "Chapter 2: an introduction to tabu search", in: F. Glover, G.A. Kochenberger (Eds.), Handbook of Metaheuristics, Kluwer Academic Publishers, 2003, pp. 37–54.
- [8] M. Fathian, B. Amiri, A. Maroosi, "Application of honey-bee mating optimization algorithm on clustering", Applied Mathematics and Computation 190 ,2007, pp.1502–1513.
- [9] Bahmani Firouzi, B., Sha Sadeghi, M., Niknam, T, "A new hybrid algorithm based on PSO, SA, and K-means for cluster analysis." International Journal of Innovative Computing Information and Control 6 (4), 2010, pp.1–10.
- [10] D. Karaboga, "An Idea Based on Honey Bee Swarm for Numerical Optimization", Technical Report TR06, Erciyes University, 2005.
- [11] K. Althoefer, "Neuro-fuzzy motion planning for robotic manipulators", Ph.D. thesis, King's College, London, 1997.
- [12] Dong Z. N. • CZhang R. L. • CChen Z. J., et al , "Study on UAV Path Planning Approach Based on Fuzzy Virtual Force." Chinese Journal of Aeronautics, 2010.
- [13] C. Mou, W. Qing-xian, J. Chang-sheng, "A modified ant optimization algorithm for path planning of UCAV", Applied Soft Computing 8 (4) ,2007, pp.1712–1718.
- [14] M. Saska, M. Macas, L. Preucil, L. Lhotska , "Robot path planning using particle swarm optimization of Ferguson splines", in: Conf. Emerging Technologies and Factory Automation, IEEE, 2006, pp.833–839.
- [15] Foo, J.L., Knutzon, J., Kalivarapu, V., Oliver, J., and Winer, E, "Path planning of unmanned aerial vehicles using B-Splines and particle swarm optimization". Journal of Aerospace Comput. Info. & Communication, 6(4), 2009, pp.271–290.
- [16] S. Piao, B. Hong, "Path planning of robot using genetic annealing algorithm", in: Proc. of the 4th World Congress on Intelligent Control and Automation, vol. 1, 2002, pp. 493–495.
- [17] M.G. Park, M.C. Lee, "Experimental evaluation of robot path planning by artificial potential field approach with simulated annealing", in: Proc. of the 41st SICE Annual Conf., vol. 4, 2006, pp. 2190–2195.
- [18] Z. N. Dong, Z. J. Chen, R. Zhou, R. L. Zhang. "A hybrid approach of virtual force and A\* search algorithm for UAV path re-planning" In Proceedings of the 6th IEEE International Conference on Industrial Electronics and Applications, IEEE, Beijing, China, 2011, pp. 1140 1145.

- [19] Melchior, N.A., Simmons, R., "Particle RRT for path planning with uncertainty." In Proceedings of the IEEE International Conference on Robotics and Automation, 2007.
- [20] S. Mittal and K. Deb, "Three-Dimensional Offline Path Planning for UAVs Using Multi objective Evolutionary Algorithms", Evolutionary Computation IEEE Conference, 2007, pp.1-6.
- [21] Hatamlou, A., "Black hole: A new heuristic optimization approach for data clustering." Information Sciences 222, 2013, pp.175-184.
- [22] Kaveh A, Talatahari S. "Optimum design of skeletal structures using imperialist competitive algorithm". Comput Struct, 2010, pp-1220–9.
- [23] SAE Aerospace Control and Guidance Systems Committee (ACGSC) Meeting Harvey's Resort, Lake Tahoe, Nevada, 2005.
- [24] D. Simon, "Biogeography-based optimization", IEEE T. Evolut. Comput., vol. 12, (2008), pp. 702–713.