A Self-organizing Multi-agent Cooperative Robotic System: An Application of Cohort Intelligence Algorithm

Palash Roychowdhury, Siddarth Mehra, Rishi Devarakonda, Prakhar Shrivastava, Siddharth Basu and Anand J. Kulkarni

Abstract This paper presents an application of the emerging Cohort Intelligence (CI) algorithm in the domain of swarm robotics. The application could be relevant to search and rescue in alien territory as well as establishment. The robots are considered as candidates in the CI algorithm. An exponential probability approach is proposed by which every candidate/robot decide to follow one another. In this approach, the probability of following the worse candidate decreases and the probability stake of the better candidate increases. This makes the robots more biased to follow better candidates. This helps to reduce the randomness in the system. The approach was applied and validated by solving path planning and obstacle avoidance for application of a swarm of robots in a static and unknown environment. The cases such as No Obstacle Case (NOC), Rectangular Obstacle Case (ROC), Multiple Rectangular Obstacles Case (MROC) and Cluttered Polygonal Obstacles Case (CPOC) were solved. The results obtained were better in terms of computational time and function evaluations as compared to the linear probability distributions approach. The limitations of the approach solving the obstacle avoidance for swarm of robots are also discussed.

P. Roychowdhury · S. Mehra · R. Devarakonda · P. Shrivastava · S. Basu · A. J. Kulkarni
Symbiosis Institute of Technology, Symbiosis International University, Pune 412115, MH, India
e-mail: palash.roychowdhury@sitpune.edu.in

S. Mehra
e-mail: siddarth.mehra@sitpune.edu.in

R. Devarakonda
e-mail: rishi.devarakonda@sitpune.edu.in

P. Shrivastava
e-mail: prakhar.shrivastava@sitpune.edu.in

S. Basu
e-mail: siddharth.basu@sitpune.edu.in

A. J. Kulkarni (✉)
Odette School of Business, University of Windsor, 401 Sunset Avenue, Windsor, ON N9B3P4, Canada
e-mail: anand.kulkarni@sitpune.edu.in; kulk0003@uwindsor.ca

A. J. Kulkarni et al. (eds.), *Socio-cultural Inspired Metaheuristics*,
Studies in Computational Intelligence 828,
https://doi.org/10.1007/978-981-13-6569-0_2

1 Introduction

According to Grayson [6], Search and Rescue (SAR) robotic systems are becoming important especially in the urban and densely populated environment. The collapsed structures, unknown establishments and hostile environment due to disasters like earthquake, tsunami, etc. add further complexity to the search and rescue operations. A single robotic system for such operation may have certain limitations including less robustness, i.e. such system is prone to single point failure. Moreover, the cost may increase with increase in number of capabilities of the robots. Multi-robotic systems are comparatively more fault tolerant with reduced communication load as well as may bring flexibility in the system [14, 15]. Although vision-based robotic systems have been deployed, most of the systems follow simple coordination rules without explicit teamwork models or goals. An Urban Search And Rescue (USAR) robotic system was developed by Burion [2]. As human intervention is required for the decision making, it adds communication overload and associated delays. This makes the integration and coordination amongst the robots remain a challenge.

The algorithm of Cohort Intelligence (CI) was proposed by Kulkarni et al. in [9]. It is inspired from the social behavior of candidates in a cohort. These candidates compete and interact with one another to enrich their independent behavior. Based on certain probability every candidate iteratively chooses another candidate to follow and chooses the values of variables/qualities in the close neighborhood from within. This process continues until there is no significant change in the behavior of all the candidates in the cohort or the goal is achieved. So far CI has been applied for solving continuous unconstrained [9] and constrained test problems [12, 13]. A modified version of CI (MCI) with improved local search ability using mutation was also developed and applied for solving several clustering problems. A hybridized version of CI and K-means was also successfully developed solving these problems [7]. The CI algorithm was further applied for solving large sized combinatorial problems from healthcare domain, a practical version of multiple knapsack problem referred to as sea-cargo mix problem (more than 25,500 variable), and selection of cross-border shipper's problem from transportation domain (more than 850,000 variables) [8, 10]. In addition, CI was applied successfully for solving 0–1 knapsack problems [11]. It is important to mention that in these earlier versions of CI, every candidate employs linear probability approach to choose a candidate to follow. The linear probability value is directly proportional to the behavior of the candidate and the probability stake associated with the roulette wheel is directly proportional to the quality of the behavior of the individual candidate. There are chances that a worse candidate may be followed by a candidate with comparatively better behavior. According to Kulkarni et al. [9], it helped the candidates jump out of local minima and reach the global minima; however, it may also increase the computational cost (time and function evaluations). Recently, CI was applied for solving practical heat exchanger design [3] as well as in hybridization with AHP [4, 5].

Along with the existing linear probability approach [9] (Kulkarni et al. 2016), in the current work presented in this paper an exponential probability version is proposed. In this approach, the probability of following the worse candidate decreases and the probability stake of the better candidate increases. This makes the robots more biased to follow better candidates. This helped to reduce the randomness in the system. This approach was applied and validated by solving path planning and obstacle avoidance of application of a swarm of robots. The results obtained from both the linear and exponential probability distributions were compared. It was observed that the time required for the robots to converge and reach the light source using the exponential probability algorithm was less than that of linear probability approach.

The further paper is organized as follows: Sect. 2 describes the application details of CI algorithm for obstacle avoidance of a swarm of robots. Section 3 provides the details on experimental evaluations of the cases such as No Obstacle Case (NOC), Rectangular Obstacle Case (ROC), Multiple Rectangular Obstacles Case (MROC) and Cluttered Polygonal Obstacles Case (CPOC). The conclusions and future directions are provided at the end of the manuscript in Sect. 4.

2 Cohort Intelligence for Swarm of Robots

The application is inspired from the role of robots in the rescue operations. The particular operation considered in this paper is associated with the robots deployed in an alien establishment with certain obstacles. The only light source is the door exit point or the target point. The robots are equipped with light sensors. The robots can communicate with one another to learn about the light intensity. The robots can do a variety of jobs such as collecting images, objects, etc. They cooperate with one another to come out of the alien establishment and reach the target point which is the light source. The mathematical formulation is discussed below.

Assume, there are N robots randomly positioned in an arena with static and unknown environment (refer to Fig. 1). Every robot is assumed to have two sensors: light sensor and proximity sensor. Every robot is represented by $B_i, i = 1, 2, \ldots, N$ receives the light with an intensity $L_i, i = 1, 2, \ldots, N$. The goal/objective is to collectively reach the light source L which is a possible exit door of the arena. In other words, the robots which are randomly located inside an establishment aims to come out of it from a door which could be considered as a light source. We assume that there is no other light source in the establishment; however, such variation with more than one light source could be modelled in a similar way. In addition, we also assume that the light source and all the robots are on the same plane. The goal could be formulated as follows:

$$Maximize \quad \sum_{i=1}^{N} L_i \tag{1}$$

The well-known inverse square law for light intensity L_i and corresponding Euclidean distance d_i is as follows: $L_i \propto 1/(d_i)^2, i = 1, 2, \ldots, N$. This motivates

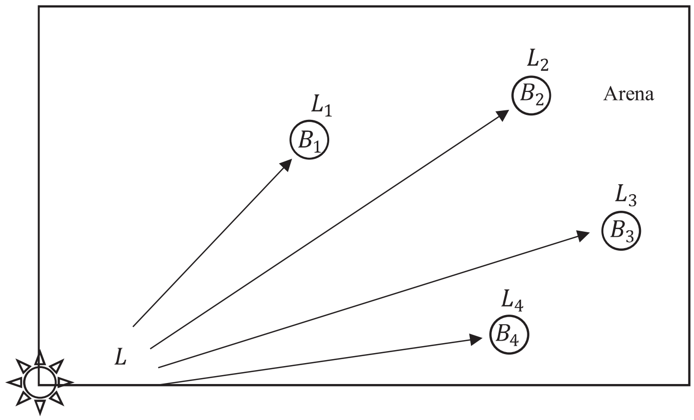

Fig. 1 An illustrative example of arena and robots

the robots to move towards the light source L. The CI procedure begins with random assignment of the initial location (x_i, y_i) of every robot $B_i, i = 1, 2, \ldots, N$, learning attempt counter $n = 1$, convergence parameter ε and maximum number of learning attempts n_{max}.

Step 1 The probability of selecting a robot to follow by any of the robot $B_i, i = 1, 2, \ldots, N$ is calculated as follows:

$$p_i^Z = L_i^Z / \sum_{i=1}^{N} L_i^Z \tag{2}$$

where z is the bias exponent.

Step 2 Every robot $B_i, i = 1, 2, \ldots, N$ generates a random number $rand \in [0, 1]$ and using roulette wheel selection approach decides to follow a certain robot.

Step 3 Every robot then moves in random direction by a step length distance d towards the corresponding robot being followed. If the robot follows itself then the robot moves away from its current location by a step length distance d. It is important to mention that the value of d is chosen based on the preliminary trials of the algorithm. Refer to Fig. 2 for details.

Step 4 If either of the following conditions is satisfied then robots stop at their current locations else go to Step 1. The change in the light intensity of every robot $B_i, i = 1, 2, \ldots, N$ does not improve significantly for considerable number of successive learning attempts, i.e. the condition $L_i^n - L_i^{n-1} \leq \varepsilon, i = 1, 2, \ldots, N$ is satisfied for considerable number of successive learning attempts. Maximum number of learning attempts n_{max} reached.

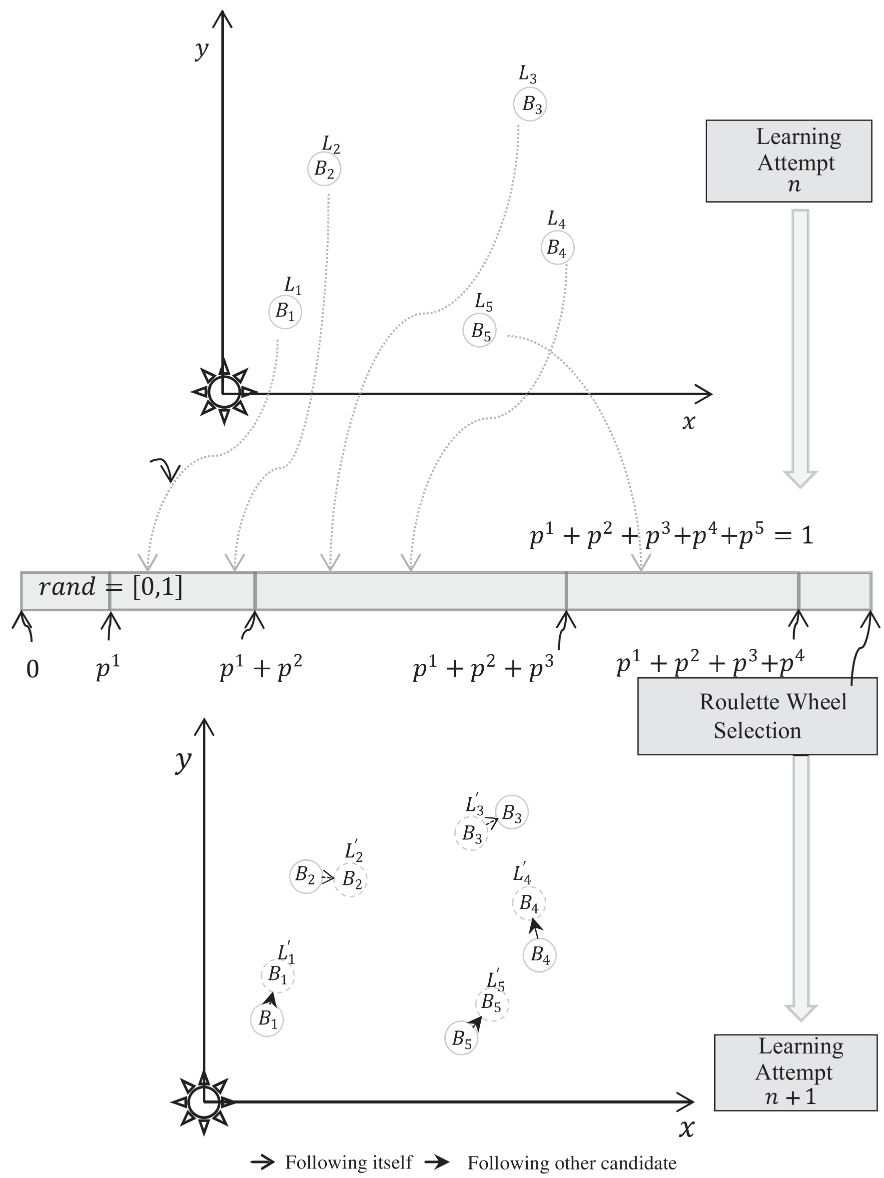

Fig. 2 Cohort intelligence framework

3 Experimental Evaluation

The Self-organization behaviour of swarm of 5 (five) robots using CI algorithm was tested on a total of 4 independent cases: No Obstacle Case (NOC), Rectangular Obstacle Case (ROC), Multiple Rectangular Obstacles Case (MROC) and Cluttered Polygonal Obstacles Case (CPOC). Twenty independent configurations (arrangement of obstacle(s), initialization of robots in the arena) for every case were generated. Every configuration/arrangement was solved 20 times using CI algorithm. The

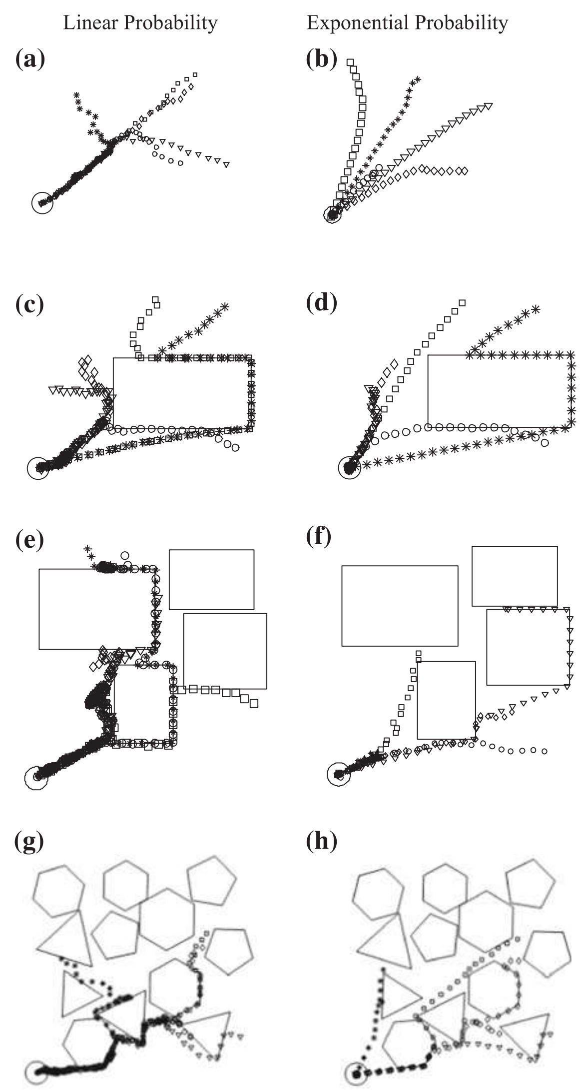

Fig. 3 Configurations: NOC, ROC, MROC and CPOC for linear and exponential probability

algorithm was coded in MATLAB R2014b and the simulations were run on Windows 8.1 platform with Intel Core i7 2.3 GHz processor speed and 8 GB RAM. The cases and the solutions are discussed below.

No Obstacle Case (NOC): In this case, there is no obstacle and the robots are located randomly in the arena. Twenty independent configurations were generated and every configuration arrangement was solved twenty times using CI with linear probability as well as exponential probability. An illustration of the NOC configuration is presented in Fig. 3a, b. The CI performance for every configuration is presented in Table 1 and Fig. 4. It could be noted that the average distance travelled by the robots as well as associated computational cost, i.e. function evaluations (FE) and time for every configuration using CI with exponential probability was significantly less when compared to the linear probability. It is important that the mean time for the robots to cluster together with CI using linear probability is less as compared to the CI with exponential probability; however, the total time required for the robots to reach the light source is less for the later approach. Moreover, the standard deviation (SD) indicated that the CI with exponential probability approach exhibited more robustness.

Rectangular Obstacles Case (ROC) and Multiple Rectangular Obstacles Case (MROC): In the case of ROC a rectangle is randomly arranged in the arena. Twenty independent configurations were generated and every configuration arrangement was solved twenty times using CI with linear probability as well as exponential probability. An illustration of the ROC configuration is presented in Fig. 3c, d. In the case of MROC different number of rectangles were randomly arranged in the arena. Twenty independent configurations were generated and every configuration arrangement was solved twenty times using CI with linear probability as well as exponential probability. An illustration of the MROC configuration is presented in Fig. 3e, f. The CI performance for every configuration is presented in Table 2 and Fig. 5. The average distance travelled by the robots using CI with exponential probability was comparable to CI with linear probability. The associated computational cost, i.e. FE and time taken for every configuration was significantly less with the earlier approach. The average time for the robots to cluster together as well as the time further to reach the light source was comparable in both the approaches; however, the total time and standard deviation (SD) indicated that the CI with exponential probability approach exhibited more robustness.

Cluttered Polygons Obstacles Case (CPOC): In this case different type of polygons were arranged closer to one another. Similar to the earlier cases, twenty independent configurations were generated and every configuration arrangement was solved twenty times using CI with linear probability as well as exponential probability. An illustration of the CPOC configuration is presented in Fig. 3g, h. The CI performance for every configuration is presented in Table 3 and Fig. 6. The average distance travelled by the robots using CI with exponential probability was significantly less as compared to the linear probability. The associated FE were significantly less; however, the average time for the robots to cluster together as well as the time

Table 1 Performance of CI with linear and exponential probability for no obstacle case (NOC)

	CI with linear probability							CI with exponential probability						
Config.	Mean (FE)	SD (FE)	Mean time (s)	SD time (s)	Mean distance travelled	Mean total time (s)	SD total time (s)	Mean (FE)	SD (FE)	Mean time (s)	SD time (s)	Mean distance travelled	Mean total time (s)	SD total time (s)
1	1067	442.95	1.30	0.80	4.02	2.03	0.91	852	370.47	1.43	0.80	3.04	1.70	0.80
2	1345	423.19	2.58	0.85	5.75	2.84	1.14	1145	192.88	2.37	0.60	4.61	2.39	0.61
3	1087	517.33	2.04	0.72	3.32	2.81	0.84	1032	411.36	2.26	0.86	3.05	2.43	0.72
4	1240	460.78	2.02	0.67	6.00	2.21	0.84	1095	217.65	2.18	0.53	5.27	2.21	0.49
5	987	425.51	2.31	0.60	4.68	2.56	0.64	947	347.80	2.04	0.59	4.19	2.17	0.64
6	1347	439.04	2.27	0.85	4.23	2.58	0.92	852	367.35	1.72	0.74	3.54	1.80	0.76
7	1115	982.74	2.69	2.38	3.98	2.76	3.09	1010	539.61	2.33	0.99	3.38	2.40	1.29
8	855	359.71	1.44	0.59	5.07	1.63	0.72	742	168.11	1.46	0.54	4.23	1.49	0.52
9	975	342.85	1.82	0.87	4.13	1.88	0.92	935	214.44	1.78	0.60	3.79	1.81	0.59
10	1192	341.24	2.91	0.69	4.77	3.26	0.72	1140	133.74	2.58	0.63	4.08	2.63	0.56
11	1505	514.46	1.54	2.40	4.17	2.48	2.60	1070	314.58	1.60	0.77	3.20	1.85	0.65
12	1287	2238.89	2.25	0.72	7.60	2.65	5.74	1180	173.49	2.46	0.60	4.17	2.49	0.57
13	1230	237.86	2.36	0.60	4.93	2.62	0.64	1022	239.65	2.28	0.52	4.19	2.41	0.57
14	1170	856.73	1.90	0.73	5.20	2.23	1.34	1092	314.05	1.77	0.74	3.96	1.85	0.70
15	1382	332.99	2.23	1.32	5.10	2.25	1.23	1147	305.99	2.07	0.93	4.06	2.08	0.93
16	1832	915.33	2.89	0.68	5.78	3.43	2.17	1147	183.23	2.46	0.55	3.49	2.49	0.55
17	1152	1051.20	2.15	1.04	4.92	2.57	2.25	1025	485.90	2.07	0.64	3.61	2.14	1.20
18	1072	438.67	1.74	0.64	4.65	2.09	0.74	910	254.16	1.62	0.43	3.80	1.64	0.43
19	1040	521.71	1.18	0.72	4.29	1.79	0.89	770	286.06	1.23	0.72	3.08	1.25	0.75
20	1242	583.85	1.57	0.72	4.51	1.96	1.04	955	1075.34	1.59	0.66	4.26	1.69	2.31

Table 2 Performance of CI with linear and exponential probability for multiple rectangular obstacles case (MROC)

	CI with linear probability							CI with exponential probability						
Config.	Mean (FE)	SD (FE)	Mean time (s)	SD time (s)	Mean distance travelled	Mean total time (s)	SD total time (s)	Mean (FE)	SD (FE)	Mean time (s)	SD time (s)	Mean distance travelled	Mean total time (s)	SD total time (s)
1	1177	465.51	0.75	0.13	4.27	1.96	0.76	517	397.35	0.88	0.14	2.47	0.98	0.64
2	1220	560.11	0.85	0.13	4.39	2.28	0.98	630	361.54	1.01	0.23	3.06	1.25	0.56
3	1660	397.33	0.71	0.17	4.30	2.52	0.76	1115	531.20	0.88	0.17	2.99	2.04	0.80
4	1290	544.73	0.44	0.08	4.37	1.18	0.50	555	272.64	0.51	0.12	2.90	0.59	0.23
5	1265	532.74	0.63	0.14	4.64	1.55	0.75	575	358.62	0.71	0.20	2.88	0.96	0.52
6	1462	685.98	0.65	0.17	4.15	1.82	0.86	602	653.97	0.77	0.18	3.19	0.97	0.85
7	1465	487.58	0.73	0.23	4.04	2.16	0.80	1155	544.03	0.82	0.14	2.89	1.70	0.73
8	740	391.20	0.78	0.14	4.55	1.16	0.54	542	96.59	0.84	0.17	3.13	0.87	0.14
9	947	497.65	0.78	0.13	3.94	1.44	0.59	557.5	247.62	0.87	0.14	2.66	0.95	0.31
10	1377	502.45	0.82	0.16	4.15	1.82	0.71	632	196.63	0.95	0.18	2.50	1.09	0.20
11	1592	353.31	0.68	0.15	4.26	2.30	0.55	1045	367.17	0.76	2.29	2.73	1.15	2.46
12	1255	406.14	0.68	0.16	4.14	1.62	0.64	580	328.41	0.79	0.22	2.285	0.98	0.40
13	1165	380.96	0.64	0.18	4.25	1.57	0.60	545	384.64	0.70	0.17	2.60	0.80	0.62
14	1465	561.29	0.36	0.05	4.46	1.08	0.40	835	487.85	0.41	0.06	3.01	0.67	0.34
15	1280	556.98	0.36	0.06	4.03	0.96	0.39	652	357.15	0.43	0.06	2.72	0.54	0.24
16	1480	342.73	0.42	0.05	4.42	1.30	0.28	652	281.88	0.55	0.07	2.30	0.63	0.23
17	1517	612.94	0.44	0.06	4.15	1.38	0.52	662	436.64	0.53	0.06	2.69	0.66	0.35
18	1260	550.15	0.47	0.10	4.31	1.13	0.46	642	429.54	0.58	0.10	3.15	0.66	0.35
19	965	601.56	0.45	0.05	3.73	0.91	0.49	502	455.07	0.50	0.05	2.70	0.57	0.36
20	1447	502.71	0.42	0.07	4.30	1.29	0.42	730	351.28	0.51	0.07	2.98	0.71	0.28

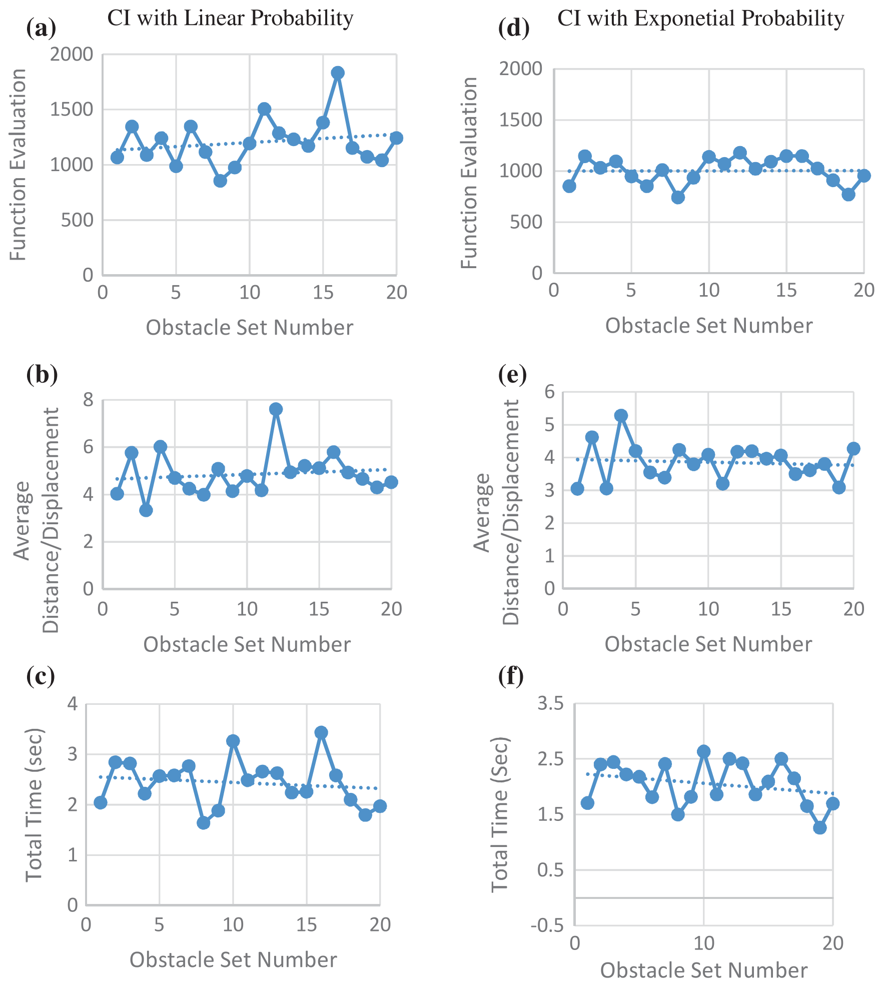

Fig. 4 Performance of CI with linear and exponential probability for no obstacle case (NOC)

to reach the light source was significantly less in the approach of CI with exponential probability as compared to the CI with linear probability. The total time SD indicated that the CI with exponential probability exhibited more robustness. It is important to note that the average time and the total time required for the robots are less in the CPOC as compared to the NOC (refer to Tables 1 and 3). This is because in case of the NOC, the light intensity associate with every robot is almost equal. This makes the individual robot learning slower. This also increases the computational cost (time and FE) for the robots to cluster together. On the other side, in case of the CPOC, due to obstacles, the light intensity associated with every robot varies considerably, which helps in increased learning and reduced computational cost.

Table 3 Performance of CI with linear and exponential probability for cluttered polygons obstacles case (CPOC)

	CI with linear probability							CI with exponential probability						
Config.	Mean (FE)	SD (FE)	Mean time (s)	SD time (s)	Mean distance travelled	Mean total time (s)	SD total time (s)	Mean (FE)	SD (FE)	Mean time (s)	SD time (s)	Mean distance travelled	Mean total time (s)	SD total time (s)
1	1085	381.58	1.18	1.89	4.70	1.73	2.01	827	350.53	1.17	0.36	3.76	1.45	0.52
2	1705	525.93	1.38	0.57	4.80	2.46	0.77	970	429.18	1.42	0.57	3.27	1.67	0.69
3	1335	529.86	1.45	1.47	4.09	2.05	1.70	770	477.73	1.34	0.29	2.91	1.48	0.59
4	1220	443.54	1.53	0.53	3.98	2.05	0.57	830	346.05	1.41	0.42	2.96	1.47	0.52
5	1212.5	369.66	1.68	0.45	4.23	2.24	0.66	810	658.66	1.47	0.38	3.40	1.57	1.29
6	1320	396.81	1.52	0.29	4.53	2.04	0.63	782	469.84	1.25	0.28	3.19	1.42	0.79
7	1245	460.44	1.35	0.36	3.84	1.92	0.92	797	457.86	1.25	0.35	3.05	1.46	0.79
8	1155	537.01	1.04	0.29	4.34	1.54	0.62	882	420.03	1.25	0.42	3.40	1.50	0.64
9	1105	373.17	1.13	0.47	3.80	1.61	0.68	782	187.49	1.21	0.82	2.64	1.23	0.84
10	1457	488.52	1.45	0.38	4.40	2.09	0.61	915	455.52	1.18	0.43	2.98	1.36	0.63
11	1012	329.10	1.73	0.70	3.83	2.09	0.80	905	201.53	1.86	0.46	3.36	1.91	0.51
12	1457	602.11	1.48	0.48	4.56	2.21	0.93	855	438.81	1.46	0.43	3.34	1.52	0.69
13	1067	609.75	1.61	0.47	4.28	2.07	1.09	840	591.33	1.48	0.66	3.64	1.80	1.26
14	1065	630.40	1.39	0.33	4.30	1.94	1.21	822	353.75	1.49	0.44	3.01	1.64	0.64
15	862	599.68	1.47	0.60	3.78	1.79	1.28	832	396.43	1.40	0.47	3.25	1.50	0.69
16	1832	669.64	1.65	0.49	4.90	3.29	1.36	1055	585.11	1.51	0.44	3.08	2.05	1.35
17	1587	465.60	1.90	0.63	4.11	2.76	0.79	1015	599.76	1.54	0.47	2.99	1.92	1.03
18	1237	650.33	1.30	0.42	4.45	2.10	1.45	820	340.38	1.51	0.43	2.87	1.64	0.51
19	1227	507.91	1.21	0.53	4.23	1.95	0.97	722	410.57	1.14	0.23	2.97	1.27	01.69
20	1430	684.84	1.25	0.61	4.18	2.21	1.54	937	538.51	1.30	0.50	2.84	1.90	0.88

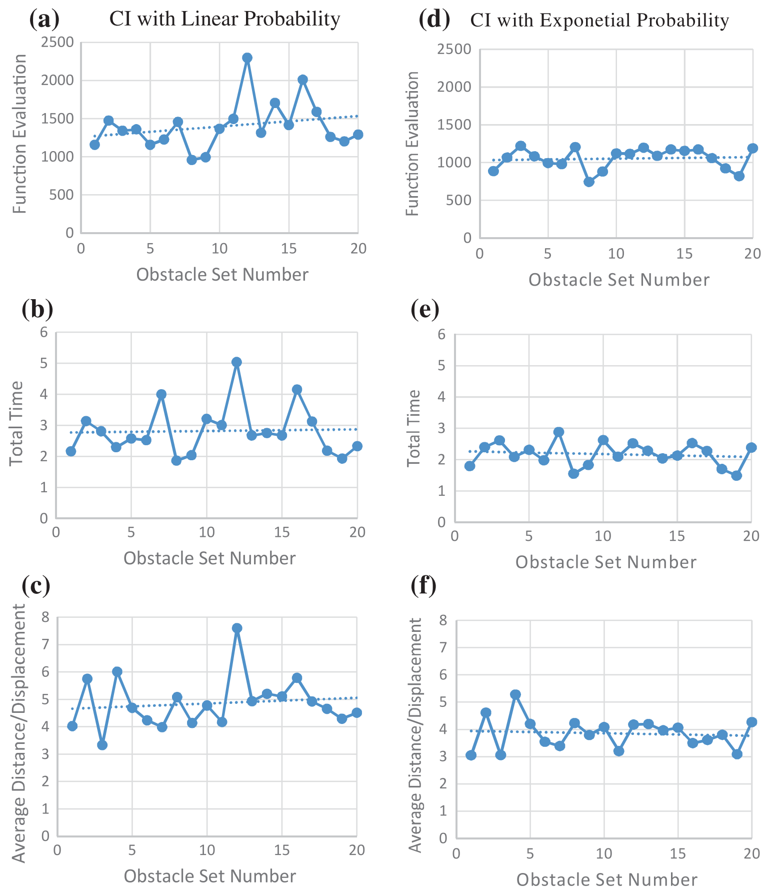

Fig. 5 Performance of CI with linear and exponential probability for multiple rectangular obstacles case (MROC)

4 Conclusions and Future Directions

The Self-organization behaviour of swarm of robots using CI algorithm was tested on a total of four independent cases: No Obstacle Case (NOC), Rectangular Obstacle Case (ROC), Multiple Rectangular Obstacles Case (MROC) and Cluttered Polygonal Obstacles Case (CPOC). The tests were conducted by randomly positioning the robots in an arena with static and unknown environment. Along with the existing linear probability approach an exponential probability version was proposed. This helped to increase biasedness and reduced the randomness in the system. This approach was applied and validated by solving path planning and obstacle avoidance of application of a swarm of robots for above cases. The results obtained from both the linear and exponential probability distributions were compared. It was observed

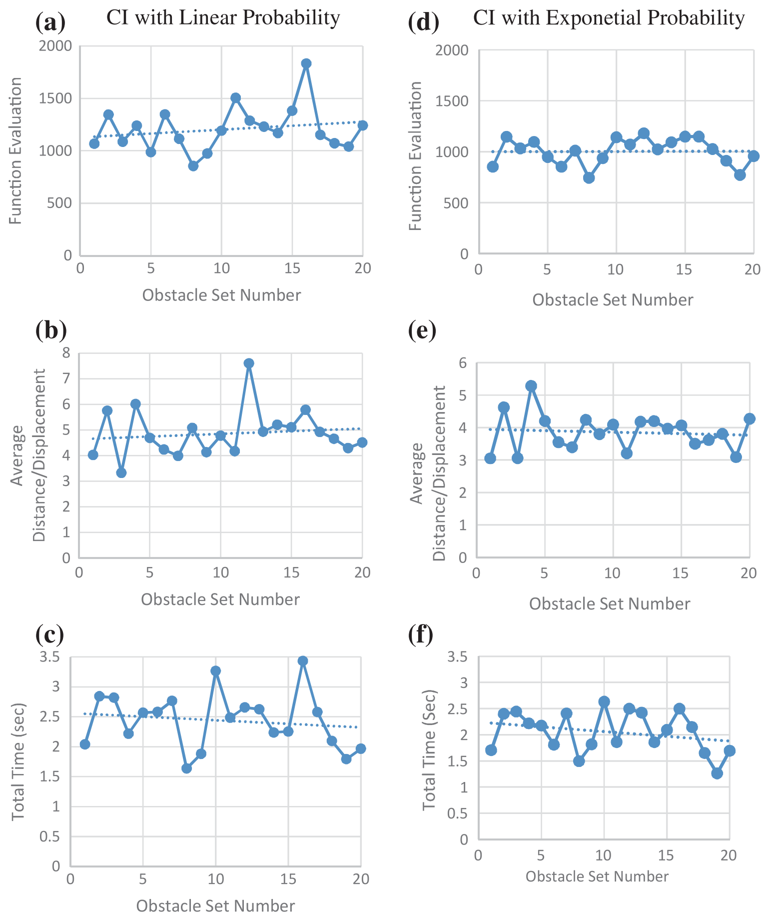

Fig. 6 Performance of CI with linear and exponential probability for cluttered polygons obstacles case (CPOC)

that the time and function evaluations required for the robots to converge and reach the light source using the exponential probability algorithm was less than that of linear probability approach. In order to maintain a balance between collective learning and individual intelligence, the exponential value z was required to be chosen based on preliminary trials. During, the preliminary trials it was observed that for higher values z, more number of robots were exhibiting individual intelligence rather than collective learning. As a result, some of the robots got stuck behind the obstacles and could not reach the target. In the near future, an approach to auto-tune such parameter needs to be addressed. Also, CI with above two approaches needs to be tested for complex U and V shaped obstacles. Authors intend to apply reinforcement learning model for robots' sapient systems [1].

References

1. Abed-alguni BH, Chalup SK, Henskens FA (2015) A multi-agent cooperative reinforcement learning model using a hierarchy of consultants, tutors and workers. Vietnam J Comput Sci 2:213. https://doi.org/10.1007/s40595-015-0045-x
2. Burion S (2004) Human detection for robotic urban search and rescue, Thesis submitted to Ecole Polytechnic
3. Dhavle SV, Kulkarni AJ, Shastri A, Kale IR (2017) Design and economic optimization of shell-and-tube heat exchanger using cohort intelligence algorithm. Neural Comput Appl (In Press)
4. Gaikwad S, Joshi R, Kulkarni AJ (2015) Cohort intelligence and genetic algorithm along with modified analytical hierarchy process to recommend an ice cream to a diabetic patient. In: Satpathy, Bhateja, Joshi (eds) Advances in intelligent and soft computing, vol 468, pp 279–288. Springer
5. Gaikwad S, Joshi R, Kulkarni AJ (2016) Cohort intelligence and genetic algorithm along with AHP to recommend an ice cream to a diabetic patient. Lecture Notes in Computer Science. Springer (In Press)
6. Grayson S (2014) Search & rescue using multi-robot systems, research report: trinity college Dublin. http://www.maths.tcd.ie/~graysons/documents/COMP47130_SurveyPaper.pdf
7. Krishnasamy G, Kulkarni AJ, Paramesaran R (2014) A hybrid approach for data clustering based on modified cohort intelligence and K-means. Expert Syst Appl 41(13):6009–6016
8. Kulkarni AJ, Baki MF, Chaouch BA (2016) Application of the cohort-intelligence optimization method to three selected combinatorial optimization problems. Europe J Op Res 250(2):427–447
9. Kulkarni AJ, Durugkar IP, Kumar M (2013) Cohort intelligence: a self supervised learning behavior. In: Proceedings of IEEE international conference on systems, man and cybernetics, pp 1396–1400. Manchester, UK, 13–16 Oct 2013
10. Kulkarni AJ, Krishnasamy G, Abraham A (2017) Cohort intelligence: a socio-inspired optimization method. In: Intelligent systems reference library, vol 114. Springer. https://doi.org/10.1007/978-3-319-44254-9, ISBN: 978-3-319-44254-9
11. Kulkarni AJ, Shabir H (2016) Solving 0–1 knapsack problem using cohort intelligence algorithm. Int J Mach Learn Cybernet 7(3):427–441
12. Kulkarni O, Kulkarni N, Kulkarni AJ, Kakandikar G (2016) Constrained cohort intelligence using static and dynamic penalty function approach for mechanical components design. Int J Parallel Emer Distrib Syst (In Press). https://doi.org/10.1080/17445760.2016.1242728
13. Shastri AS, Jadhav PS, Kulkarni AJ, Abraham A (2016) Solution to constrained test problems using cohort intelligence algorithm. In: Advances in intelligent and soft computing, vol 424, pp 427–435. Springer
14. Trigui S, Koubaa A, Cheikhrouhou O, Youssef H, Bennaceur H, Sriti M-F, Javed Y (2014) A distributed market-based algorithm for the multi-robot assignment problem. Proc Comput Sci 32:1108–1114
15. Yan Z, Jouandeau N, Cherif AA (2013) A survey and analysis of multi-robot coordination. Int J Adv Rob Syst 10:1–18