Robotica (2022), **40**, pp. 3952–3975
doi:10.1017/S0263574722000698

CAMBRIDGE UNIVERSITY PRESS

RESEARCH ARTICLE

Trajectory planning and control of multiple mobile robot using hybrid MKH-fuzzy logic controller

Saroj Kumar* and Dayal R. Parhi

Robotics Laboratory, National Institute of Technology, Rourkela, Odisha 769008, India
*Corresponding author. E-mail: Saroj4sks@gmail.com

Received: 29 January 2022; **Revised:** 6 April 2022; **Accepted:** 8 May 2022; **First published online:** 2 June 2022

Keywords: autonomous robot, control, path planning, fuzzy logic

Abstract
Robotics with artificial intelligence techniques have been the center of attraction among researchers as it is well equipped in the area of human intervention. Here, the krill herd (KH) optimization algorithm is modified and hybridized with a fuzzy logic controller to frame an intelligent controller for optimal trajectory planning and control of mobile robots in obscure environments. The controller is demonstrated for single and multiple robot's trajectory planning. A Petri-net controller has also been added to avoid conflict situations in multi-robot navigation. MATLAB and V-REP software are used to simulate the work, backed with real-time experiments under laboratory conditions. The robots efficiently achieved the goals by tracing an optimal path without any collision. Trajectory length and time spent during navigation are recorded, and a good agreement between the results is observed. The proposed technique is compared against existing research techniques, and an improvement of 14.26% is noted in terms of path length.

1. Introduction

In the past few years, the development of robotic applications has gained tremendous growth in humanoids, wheeled mobile robots, and prostheses. This development has attracted many researchers worldwide to concentrate on artificial intelligence (AI) techniques. With the development of human working gaits to robot path planning, numerous methods have made locomotion easier. While optimization techniques have paved the way to obtaining the best path to reach the target, it has also successfully smoothened the traverse path. Several research works have been published in the past few years related to the path outlining of mobile robots using various artificial techniques. Gandomi & Alavi [1] have introduced krill herd (KH) for optimization problems, in which simulation works are carried out and compared among them. Abualigah et al. [2] have proposed this optimization technique for solving clustering problems. Rao et al. [3] have presented krill- herd optimization for navigational control of wheeled robots. Singh & Thongam [4] have used fuzzy logic technique for the navigation of mobile robot in static environments. Chen et al. [5] have proposed fuzzy logic for wall following wheeled robot. Ben & Seddik [6] have proposed PID tuned fuzzy logic for the control of robot. Muni et al. [7, 8] have worked on controlling legged robots using hybrid fuzzy methods. Kumar et al. [9, 10] have compared different methods towards path optimization with a developed fuzzy-whale optimization approach in a cluttered stationary and moving obstacles environment. Apart from KH optimization and fuzzy logic technique, many other techniques are available for the control of mobile robots. For example, Mohanty et al. [11] have presented their research on adequate path planning for mobile robots using the cuckoo search technique. The work focused on identifying the best optimal path for the locomotion of robots in an obscure scenario. Patle et al. [12] discussed matrix binary-coded algorithms for robot trajectories. Recently, Kumar et al. [13] have presented hybridized optimization technique for path optimization of multiple mobile robots in obscure scenarios. Singh et al. [14] worked on the path optimization of mobile robot using an

Pseudocode: Modified Krill-Herd Optimization

Input : DOLO, DORO, DOFO
Initialize : S_i, N_i, F_i, D_i
IF Robot heads towards target **THEN**
  Calculate and generate $\frac{dS_i}{dt} = N_i + F_i + D_i$
END IF locomotion of krill is started **THEN**
  Update and feed $N_i^{New} = N^{\max} l_i + \omega_n N_i^{old}$
  Find Direction of $'i^{th}'$ krill $l_i = l_{i(present)} + l_{i(target)}$
END IF Food location is marked **THEN**
  Calculate $F_{i,\text{mod}} = V_f \beta_i + \omega_f F_i^{old}$
  Calculate $\beta_i = \beta_i^{food} + \beta_i^{best}$
END IF Random diffusion of krill not obtained **THEN**
  Calculate $D_i = D^{\max} \square \delta$
END IF Optimal point not updated **THEN**
  Calculate updated position $F_{i,\text{mod}} = \left(V_f \beta_i + \omega_f F_i^{old}\right)\left(1 - \frac{i^{th}\ iter}{\max - iter}\right)$
End
WHILE maximum iteration not attained
  DO $D_{i,\text{mod}} = D^{\max} \square \delta \left(1 - \frac{i^{th}\ iter}{\max - iter}\right)$
  Update final optimal point and calculate $S_i(t + \Delta t) = S_i(t) + \Delta t \frac{dS_i}{dt}$
  And Calculate **IPA** according to global optima
END WHILE

Figure 1. *Pseudocode for MKH optimization technique.*

artificial neural network approach. Parhi et al. [15] have worked on the controller for precise locomotion of the wheeled robot in an unknown environment. Pandey et al. [16, 17] and Mohanty et al. [18] have worked on wheeled robot navigation in an obscure workspace. Kim et al. [19] have presented an ant colony optimization technique for the loading balancing problem. Parhi et al. [20–22] have presented different AI techniques for the traversal path of robots. Cruz et al. [23] have expressed their research on mobile robots using artificial bee colony optimization techniques. Fen et al. [24] have presented an improved ACO technique for the analysis of problems. Muni et al. [25–27] have developed fuzzy and water cycle approaches for navigational control of a humanoid robots. Kumar et al. [28] have proposed a hybrid model for trajectory planning of mobile robots. Montiel et al. [29] have explained bacterial foraging behavior for path planning of robots. Ahmed et al. [30] have proposed space deformation-based motion planning of mobile robots. Bolaji et al. [31] have proposed a review on different AI techniques,

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

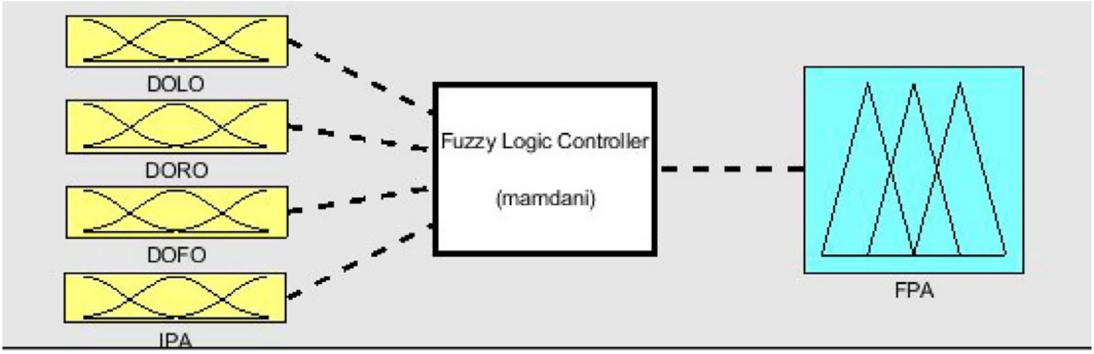

Figure 2. *Fuzzy logic controller model.*

including the KH optimization technique. Kumar et al. [32, 33] have proposed metaheuristic approaches for trajectory outlining of robots. Wang et al. [34] have proposed dynamic environment path planning using Fuzzy-artificial potential field approach. Dirik et al. [35] have proposed vision-based global path planning of four-wheeled robots. Type-2 Fuzzy interface system has been used for performance analysis. Luo et al. [36] have proposed an improved ant colony approach for path planning of mobile robots. Authors have introduced dynamic punishment approach for solving deadlock problems. Li et al. [37] have proposed Fuzzy- torque approach for lateral stability of robots and accurate trajectory planning. Teli and Wani [38] have presented autonomous navigation of robot by avoiding local optima. The problem statement has been solved using fuzzy-based approach. Luan & Thinh [39] have proposed a hybrid GA (Genetic algorithm) approach for global path outlining of the wheeled robot. Authors have improved the GA by dynamic mutation rate and fluctuating local- global approach. Kim et al. [40] have proposed UAV-assisted mobile robot navigation in a cluttered workspace. The robot has been used for 3D data collection during surveillance and topographical work. Hu et al. [41] have proposed an approach for navigational control of robots in 3D rough terrains. A sim to real pipeline training pattered has been used for controlling the robot.

After perusal of the above-cited papers, it is observed that apart from path planning, a multi-objective technique is still needed in the present scenario. This manuscript discusses about the hybrid controller of modified Krill-Herd optimization and Fuzzy logic approach (MKH-Fuzzy) to accomplish a multi-objective optimized path planning method for mobile robots in unknown terrains. Multi-objectives comprise route outlining, time optimization, smooth navigation, and avoidance of local optimal points. All the objectives are encountered through the hybrid technique. Layout of the manuscript is as follows:

In Section 2, modified MKH optimization approach is discussed. In Section 3, the fuzzy-logic controller is elaborated. The hybridization model of both techniques are discussed in Section 4. In Section 5, Petri-net controller is presented. Simulation and experimental analyses are carried out using hybrid MKH-FLC in Section 6. The proposed controller is compared against existing technique, and the details is presented in Section 7. Conclusion and future scopes are presented in Section 8.

2. Modified Krill-Herd optimization approach

The krill-herd optimization technique is a bio-inspired continuous optimization technique [31], which is well known to optimize problems. The word krill refers to the small fishes, and the krill's group is called a herd. The optimization approach deals with the hunting style of krill in an ocean. They used to chase food sources in a herd by maintaining communication with each other, and this hunting style is the motivation of the algorithm. A $'n'$dimensional search space is considered with random generation of $'P_n'$ number of krills. The individual krill position vector is initialized as per the following equation [3].

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

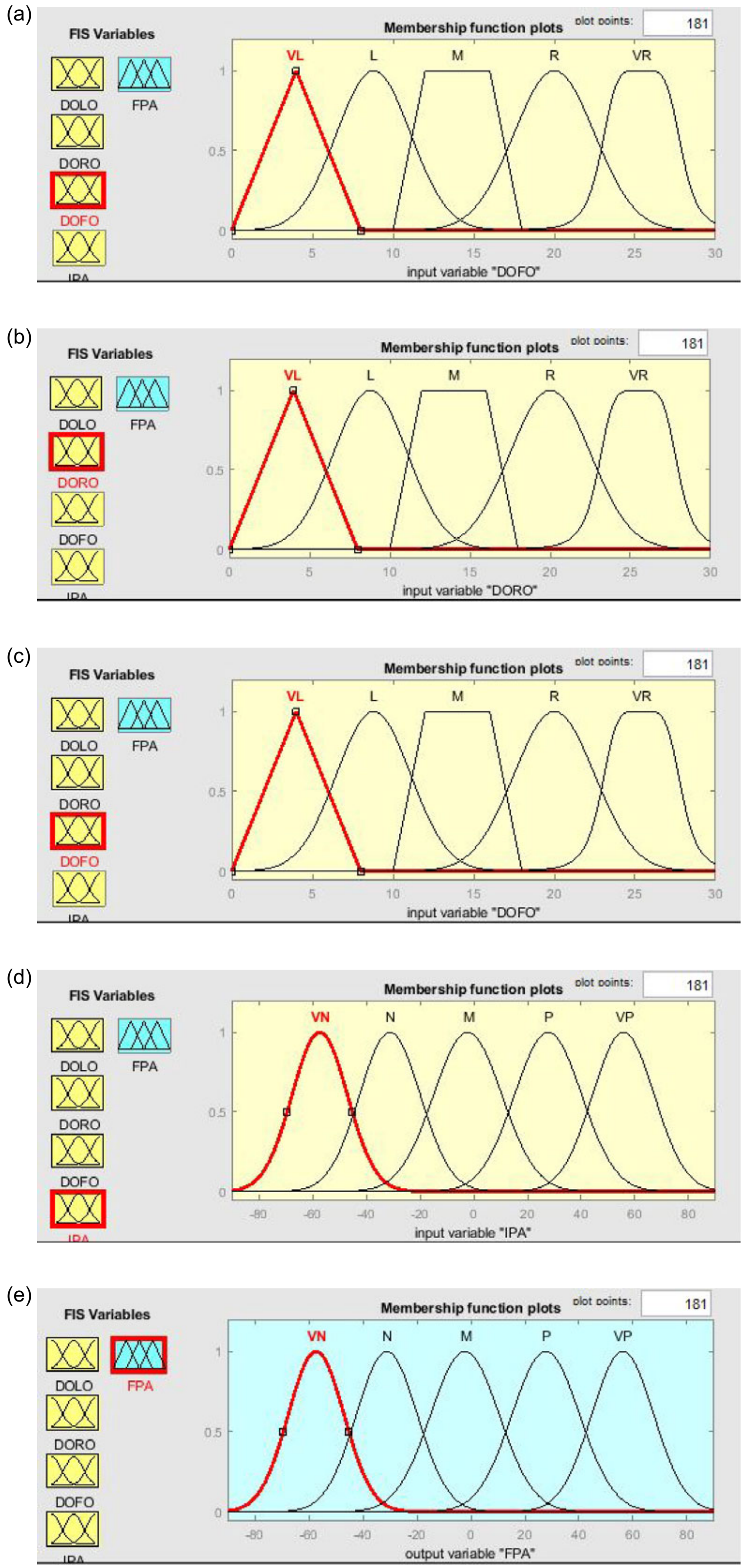

Figure 3. *Fuzzy membership function for inputs and output.*

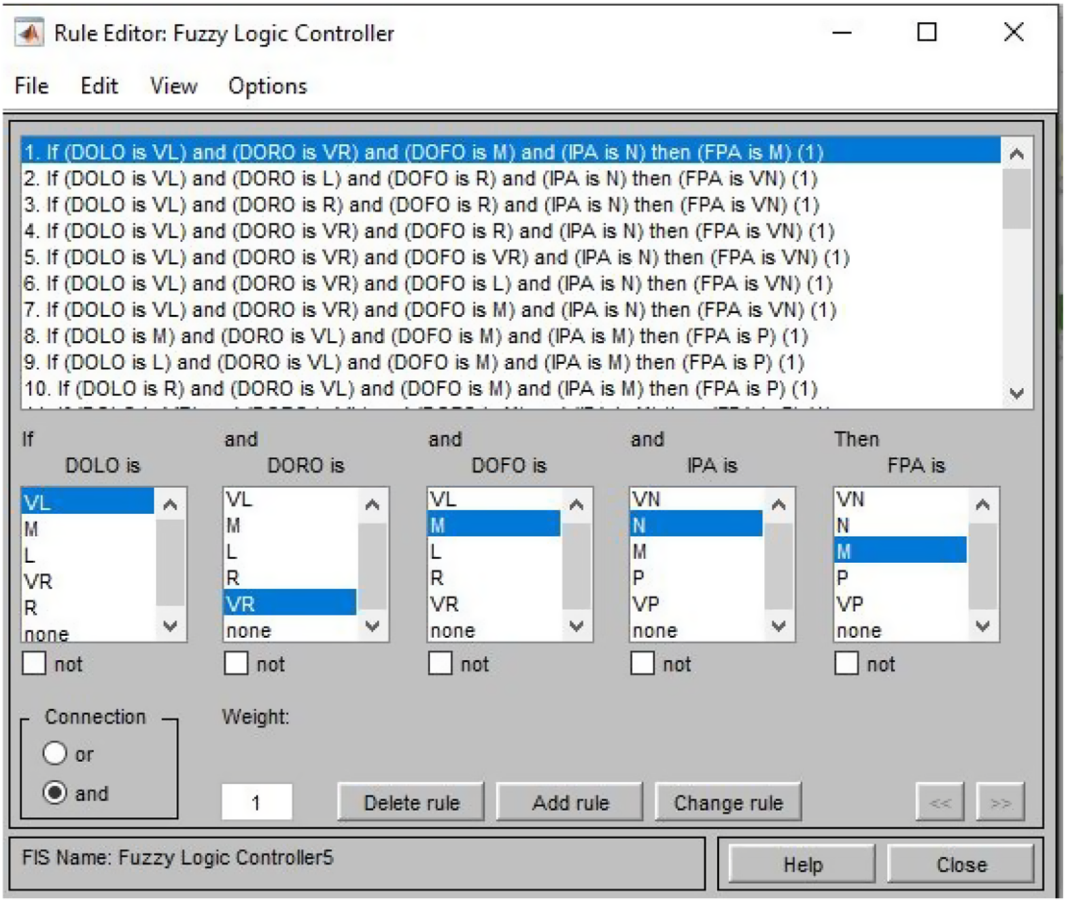

Figure 4. *Rules base.*

$$T_i^j = T_{\min}^j + rand_j \left(T_{\max}^j - T_{\min}^j\right) \tag{1}$$

The maximum and minimum limit of search space is denoted as $'T_{\max}^n'$ and $'T_{\min}^n'$, which is in $j \in [1, 2, 3 \ldots\ldots\ldots n]$. The movement of krill is an essential part of this optimization, as it searches the location of food so that the herd can find a prey source. Let the ocean be an n-dimensional space. Through "Lagrangian" approach, the movement of krill-herd is mathematically formulated as [24];

$$\frac{dS_i}{dt} = N_i + F_i + D_i \tag{2}$$

Where $'S_i'$ is the original motion of the i^{th} krill. $'N_i'$ is the motion influenced by closest krill. $'F_i'$ is foraging locomotion and $'D_i'$ is randomly selected diffusion of the krill. These above parameters are required prime attention to find an optimal location of food source, So that is formulated as per the following smooth calculation of the krill locomotion happens which are as follows:

- Krill movement with each other (N_i).
- Searching or foraging motion (F_i).
- Random diffusion of Krills (D_i).

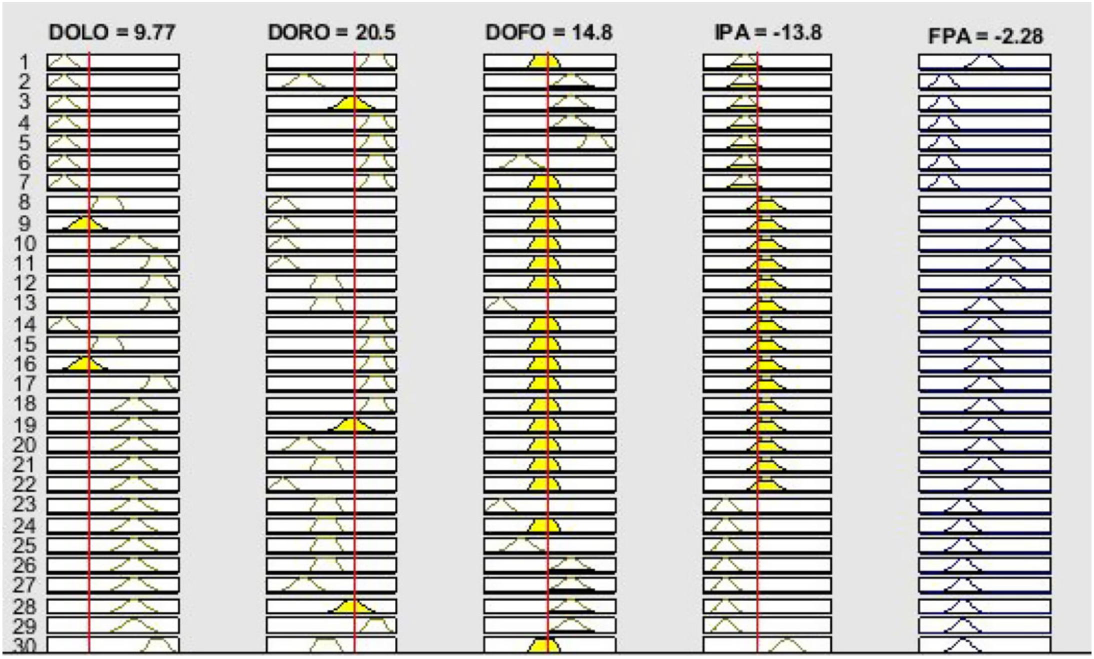

Figure 5. *Output rule generator.*

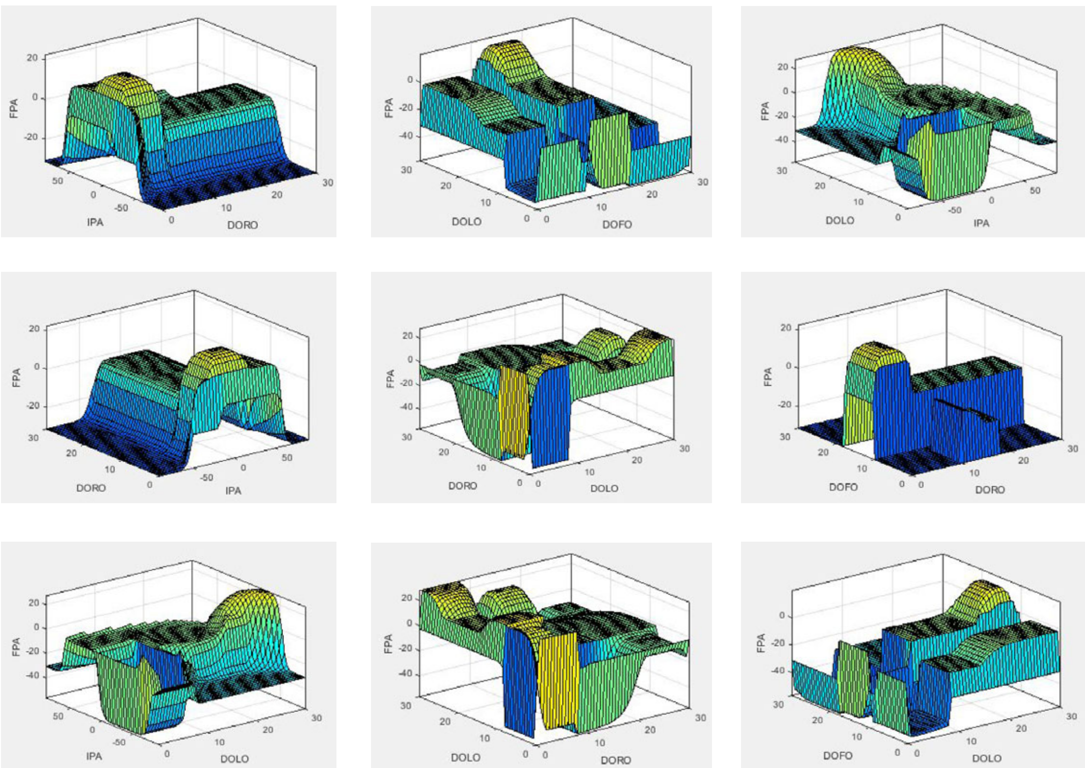

Figure 6. *Surface plot.*

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

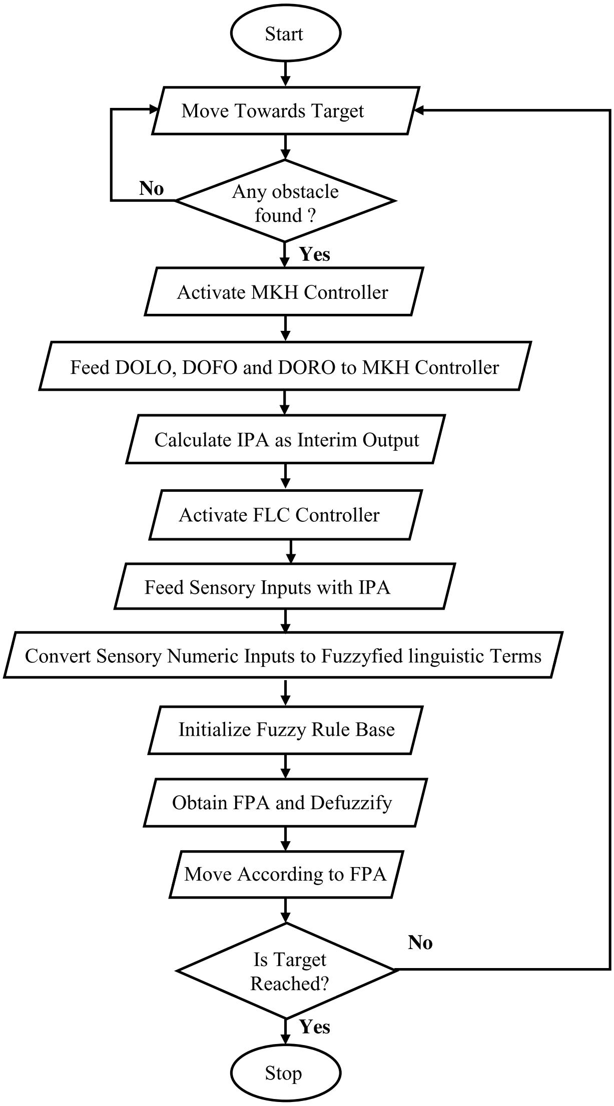

Figure 7. *Flowchart of proposed hybrid controller.*

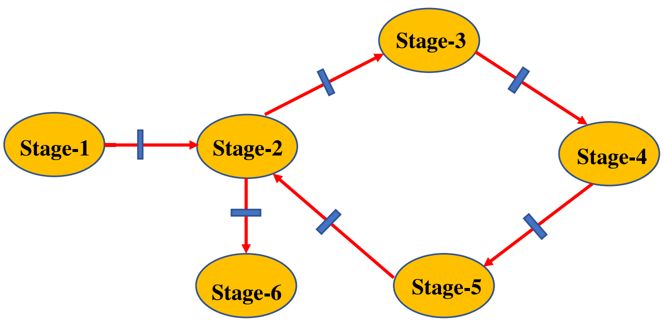

Figure 8. *Petri-net Network [7].*

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

Processor	Motorola 6833, 25MHz
RAM	512kb
Motion	2DC brushed servo motors with encoders.
Sensors	8 infrared proximity and ambient light sensors (100mm Range)
Communication	Standard serial port, upto 115kb/s
Size	Dia- 70mm, height- 30mm
Weight	80g
Payload	Max 250g.

Figure 9. *Description of K-II robot.*

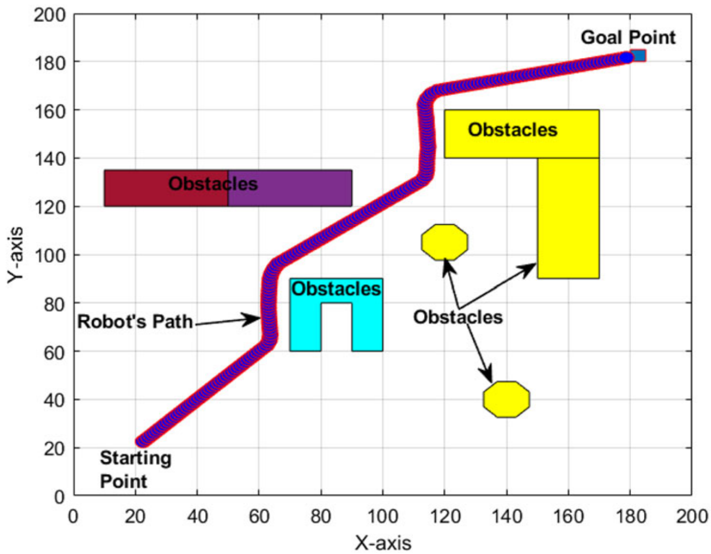

Figure 10. *Simulation analysis on MATLAB platform.*

- **The individual Krill locomotion may be represented as:**

Before moving to herd movement calculation, it is necessary to find $'N_i'$; therefore mathematical calculation is given in Eq. (3).

$$N_i^{New} = N^{\max} l_i + \omega_n N_i^{old} \tag{3}$$

Where $'N^{\max}'=$ Maximum motion-induced. $'\omega_n'=$ Inertia weight [0,1]. $'N_i^{old}'=$ Last induced motion and $'l_i'=$ Direction of navigation and it can be calculated as

$$l_i = l_{i(present)} + l_{i(target)} \tag{4}$$

These two parameters of $'l_i'$are called the local effect of Krill, and these effects occur during the movement of bulk krill and target location of prey.

- **The foraging motion of Krill-Herd is depends on 2-parameters:**

  - Location of the food.
  - Prior knowledge of food location.

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

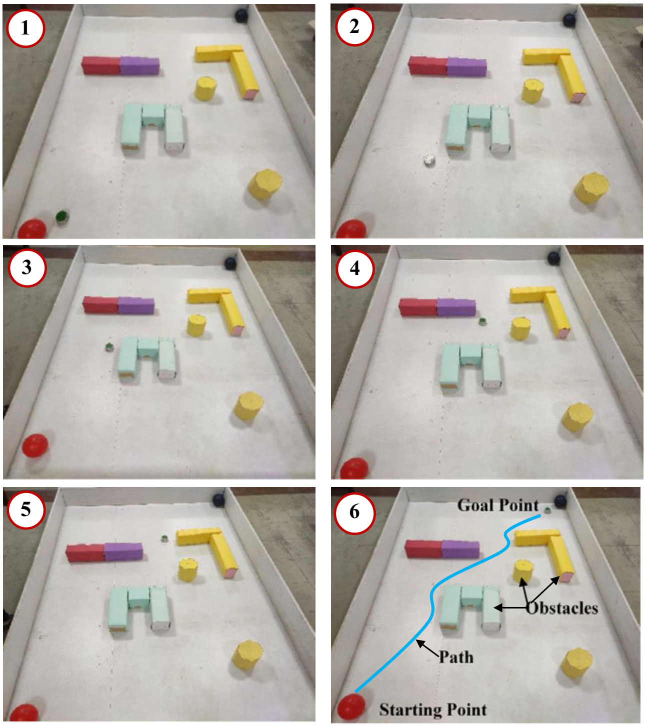

Figure 11. *Real-time experimental analysis.*

The foraging movement of krill is formulated as follows;

$$F_{i,\,\mathrm{mod}} = V_f\beta_i + \omega_f F_i^{old} \tag{5}$$

Where $'V_f'$= foraging speed (=0.02), $'\beta_i'= i^{th}$ position of krill, $'\omega_f'$= inertia weight (= [0,1]), and $'F_i^{old}'$= last foraging motion. The i^{th} position of krill can be calculated as;

$$\beta_i = \beta_i^{food} + \beta_i^{best} \tag{6}$$

$'\beta_i^{food}'$ denotes the attracting parameters for food and $'\beta_i^{best}'$ denotes best position of i^{th}krill.

- **The random diffusion can be calculated as:**

$$D_i = D^{\max}\delta \tag{7}$$

Where $'D^{\max}'$= maximum diffusion speed, and $'\delta'$= random directional vector.

2.1. Modification of Krill-Herd (MKH) optimization technique

The primary objective of route outlining is to determine a smooth optimal path. The parametric values are chosen randomly in basic KH technique, which leads to a delay in convergence time. Therefore, the parameters are modified intelligently and implemented for optimal path search to provide IPA. The modified foraging motion is represented with linearly decreasing terms in [−1, 1] in Eq. (8).

$$F_{i,\,\mathrm{mod}} = \left(V_f\beta_i + \omega_f F_i^{old}\right)\left(1 - \frac{i^{th}\ iter}{\max - iter}\right) \tag{8}$$

Where $'i^{th}\ iter'$is the value at i^{th} iteration, and $'\max - iter'$ is the maximum iteration value. In addition, the linearly decreasing term is added with a random defusing motion for fast convergence, which is

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

Table I. *Path length in experimental analysis.*

	Path length (in cm)		
Run	**In simulation**	**In experiment**	**% Deviation**
1	304.36	320.14	4.93
2	304.15	317.42	4.18
3	303.41	317.58	4.46
4	302.46	317.25	4.66
5	302.14	317.31	4.78
6	302.48	317.17	4.63
7	302.73	317.17	4.55
8	302.42	317.23	4.67
9	302.16	317.52	4.84
10	302.43	317.14	4.64
Average =	**302.87**	**317.59**	**4.63**

Table II. *Timespan in experimental analysis.*

	Timespan (s)		
Run	**In simulation**	**In experiment**	**% Deviation**
1	28.74	30.12	4.58
2	27.73	29.42	5.74
3	27.64	29.16	5.21
4	27.24	28.42	4.15
5	27.65	28.53	3.08
6	27.65	28.54	3.12
7	27.36	28.67	4.57
8	27.38	28.39	3.56
9	27.41	28.42	3.55
10	27.23	28.47	4.36
Average =	**27.60**	**28.81**	**4.19**

shown in Eq. (9).

$$D_{i,\ \mathrm{mod}} = D^{\max}\delta\left(1 - \frac{i^{th}\ iter}{\max - iter}\right) \tag{9}$$

After maximum iteration, $'i^{th}'$ krill updates its position to new global optima, which is calculated as

$$S_i\,(t + \Delta t) = S_i\,(t) + \Delta t\frac{dS_i}{dt} \tag{10}$$

$'\Delta t'$ represents the time interval that depends on the environmental condition of robot, and it is expressed as;

$$\Delta t = V_i \sum_{p=1}^{x} \left(T_{\max}^{x} - T_{\min}^{x}\right) \tag{11}$$

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

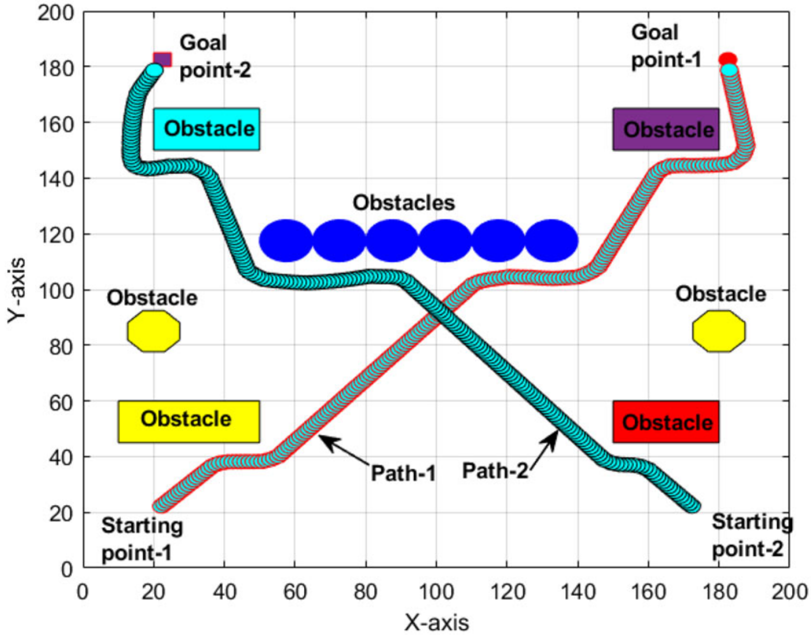

Figure 12. *Navigational analysis in scene-1.*

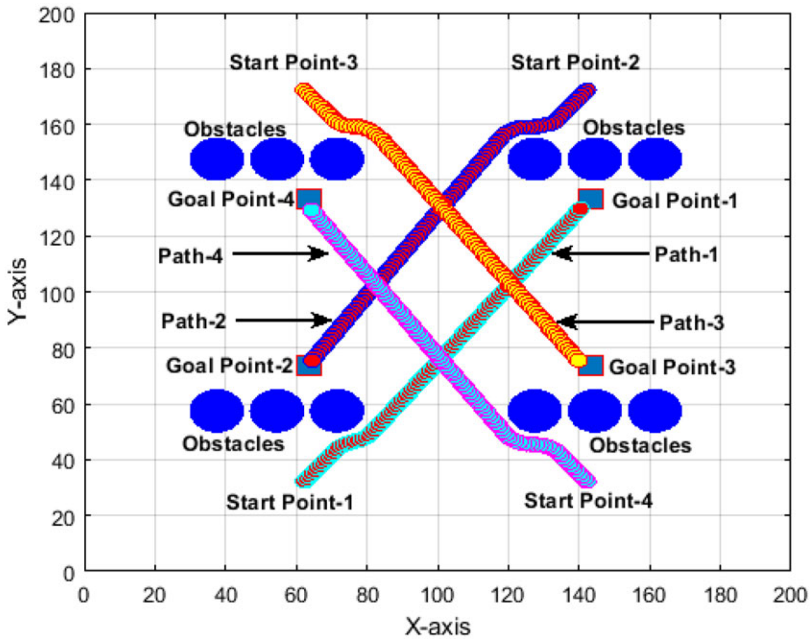

Figure 13. *Navigational analysis in scene-2.*

Where $'T^x_{\max}'$ and $'T^x_{\min}'$ denotes the maximum and minimum limit of $'p^{th}'$ variable dimension [$p \in (1, 2, 3, 4\ldots\ldots x)$] respectively. $'V_i'$ is a constant in [0,1] and used for krills to provide safe position of locomotion. The pseudocode for modified krill-herd optimization is depicted in Fig. 1, which shows the process of obtaining solutions.

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

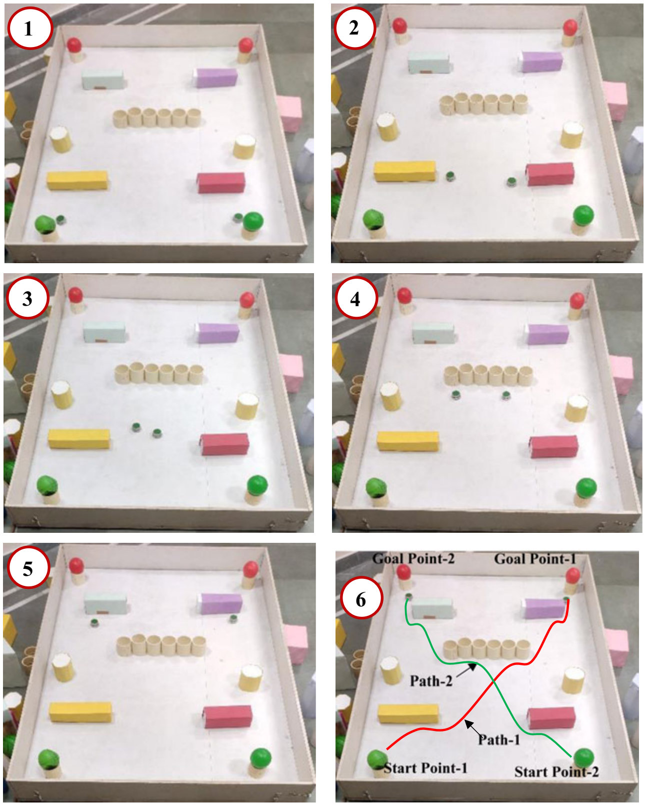

Figure 14. *Real-time experimental analysis with two robots in scene-1.*

3. Fuzzy logic controller (FLC)

Fuzzy logic is one of the simplest controllers to solve significant area problems. The human behaviour of reasoning inspires it. The controller works on rules (*IF-THEN* rules) that are framed to train the controller. While solving the problems, the FLC progressed through a number of steps, such as fuzzification of inputs implies conversation of numeric value to fuzzy code, rule generation, and defuzzification of outputs. The used model is shown in Fig. 2.

In this model, four inputs are considered such as 'DOLO' (Distance of left obstacles), 'DORO' (Distance of right obstacles), 'DOFO' (Distance of front obstacles) and 'IPA' (Initial Piloting angle); however, one output is generated as 'FPA' (Final Piloting angle). The range of inputs is considered from 0 to 30, and the range of piloting angle is -90 to $+90$ degrees. A rule base of 200rules is framed to implement the FLC. The inputs and output membership functions are shown in Fig. 3.

While designing FLC, the variables of robot navigation (membership function) are carefully included in input and output. There are five variables are considered such as 'VL' (Very left), 'L' (left), 'M' (Medium), 'R' (Right), 'VR' (Very right) for distance inputs, and 'VN' (Very Negative), 'N' (Very negative), 'M' (Medium), 'P' (Positive), 'VP' (Very positive) are considered for angle output.

The framed rules and output generation is shown in Figs. 4 and 5. The relation between input and output is shown through the surface plot in MATLAB, shown in Fig. 6.

Let the input and output membership variables DOLO, DORO, DOFO, IPA, and FPA be symbolized as 'L', 'R', 'F', 'A_1' and 'A_2' respectively. 'x', 'y', and 'z' are the membership adjusting constant. The

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

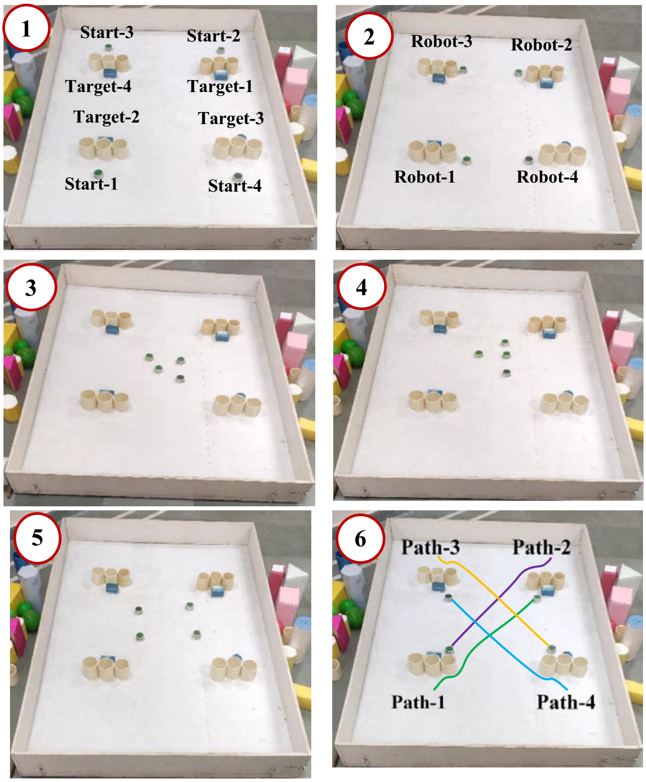

Figure 15. *Real-time experimental analysis with four robots in scene-2.*

fuzzification of input and output variables are expressed as [10]:

$$\eta_1(L) = \frac{1}{1 + \left[\frac{L - z_1}{x_1}\right]^{2y_1}} \tag{12}$$

$$\eta_2(R) = \frac{1}{1 + \left[\frac{R - z_2}{x_2}\right]^{2y_2}} \tag{13}$$

$$\eta_3(F) = \frac{1}{1 + \left[\frac{F - z_3}{x_3}\right]^{2y_3}} \tag{14}$$

$$\eta_4(A_1) = \frac{1}{1 + \left[\frac{A_1 - z_4}{x_4}\right]^{2y_4}} \tag{15}$$

The weighted average method is used to calculate the defuzzified value of output (${A_2}^*$), shown in Eq. (16).

$${A_2}^* = \sum \frac{\eta_1(L) \cdot \eta_2(R) \cdot \eta_3(F) \cdot \eta_1(A_1) \cdot A_2}{\eta_1(L) \cdot \eta_2(R) \cdot \eta_3(F) \cdot \eta_1(A_1)} \tag{16}$$

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

Table III. Path length in scene-1.

	Path length					
	In simulation		In experiment		% Deviation	
Run	R-1	R-2	R-1	R-2	R-1	R-2
1	308.42	312.42	318.7	325.5	3.23	4.02
2	308.57	312.76	318.4	325.4	3.09	3.88
3	308.76	312.43	317.5	324.8	2.75	3.81
4	308.45	312.67	317.4	324.6	2.82	3.68
5	308.76	312.53	317.8	324.8	2.84	3.78
6	308.76	312.74	317.3	324.7	2.69	3.68
7	308.69	312.51	317.8	324.7	2.87	3.75
8	308.35	312.43	317.9	324.7	3.00	3.78
9	308.64	312.59	317.6	324.8	2.82	3.76
10	308.34	312.18	317.5	324.7	2.89	3.86
Average	308.57	312.53	317.79	324.87	2.90	3.80

Table IV. Time consumption in scene-1.

	Time consumption					
	In simulation		In experiment		% Deviation	
Run	R-1	R-2	R-1	R-2	R-1	R-2
1	27.85	28.14	28.75	29.42	3.13	4.35
2	27.76	28.17	28.74	29.31	3.41	3.89
3	27.84	28.24	28.68	29.37	2.93	3.85
4	27.65	28.46	28.76	29.42	3.86	3.26
5	27.47	28.41	28.74	29.42	4.42	3.43
6	27.86	28.17	28.81	29.27	3.30	3.76
7	27.56	28.34	28.82	29.51	4.37	3.96
8	27.49	28.37	28.49	29.34	3.51	3.31
9	27.81	28.41	28.52	29.42	2.49	3.43
10	27.56	28.17	28.75	29.25	4.14	3.69
Average	27.69	28.29	28.71	29.37	3.56	3.69

4. Hybrid MKH-fuzzy controller model

The aim of developing a hybrid controller is to overcome the limitations of standalone algorithms. Robot navigation and trajectory optimization have always remained as one of the challenging research that requires accurate navigational parameters along with fast convergence. The basic advantage of classical method is fast convergence rate, whereas the reactive approach gives optimal value. Therefore, the combination of classical practice and reactive technique is developed and implemented on robots. In this model, two stages of hybridized is proposed. The initial inputs DORO, DOLO, and DOFO (Sensory information) are fed to the MKH model, and the interim output is considered as the first output. In the second phase, the output (IPA) from MKH model and sensory information (DORO, DOLO, DOFO) are fed to FLC. Further, the FLC calculates the final piloting angle (FPA) used by the robot to escape obstacles and achieve target. The hybrid model of FLC is shown in Fig. 2. The flow chart of the hybrid model is shown in Fig. 7.

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

Table V. *Path length in scene-2.*

	Path length (cm)											
	In simulation				In real-time experiment				% Deviation			
Run	**R-1**	**R-2**	**R-3**	**R-4**	**R-1**	**R-2**	**R-3**	**R-4**	**R-1**	**R-2**	**R-3**	**R-4**
1	137.17	142.56	136.78	146.24	143.2	148.6	142.2	152.2	4.21	4.06	3.81	3.92
2	137.25	142.42	136.65	146.35	142.3	148.4	142.6	152.8	3.55	4.03	4.17	4.22
3	137.41	142.18	136.91	146.72	143.6	148.3	143.3	152.5	4.31	4.13	4.46	3.79
4	136.42	142.37	136.19	146.35	143.2	148.2	142.6	152.6	4.73	3.93	4.50	4.10
5	137.56	142.54	136.87	146.68	142.3	148.3	142.8	152.3	3.33	3.88	4.15	3.69
6	137.84	142.63	136.74	146.87	142.6	148.3	142.7	152.5	3.34	3.82	4.18	3.69
7	137.26	142.34	136.62	146.35	142.6	148.6	142.8	152.5	3.74	4.21	4.33	4.03
8	137.46	142.62	136.37	146.39	142.5	148.2	142.6	152.4	3.54	3.77	4.37	3.94
9	137.28	142.6	136.42	146.43	142.3	148.2	142.6	152.5	3.53	3.78	4.33	3.98
10	137.29	142.71	136.82	146.52	142.4	148.3	142.7	152.4	3.59	3.77	4.12	3.86
Avg.	137.29	142.50	136.64	146.49	142.70	148.34	142.69	152.47	3.79	3.94	4.24	3.92

Table VI. *Time consumption in scene-2.*

	Time consumption (s)											
	In simulation				In real-time experiment				% Deviation			
Run	**R-1**	**R-2**	**R-3**	**R-4**	**R-1**	**R-2**	**R-3**	**R-4**	**R-1**	**R-2**	**R-3**	**R-4**
1	11.45	12.72	12.01	13.12	11.91	13.26	12.57	13.58	3.86	4.07	4.46	3.39
2	11.37	12.79	12.11	13.16	11.84	13.27	12.42	13.52	3.97	3.62	2.50	2.66
3	11.42	12.71	12.09	13.15	11.86	13.21	12.52	13.62	3.71	3.79	3.43	3.45
4	11.37	12.69	12.12	13.17	11.89	13.29	12.51	13.65	4.37	4.51	3.12	3.52
5	11.32	12.73	12.11	13.14	11.81	13.26	12.57	13.67	4.15	4.00	3.66	3.88
6	11.46	12.78	12.09	13.14	11.79	13.24	12.54	13.58	2.80	3.47	3.59	3.24
7	11.43	12.76	12.13	13.15	11.81	13.25	12.53	13.57	3.22	3.70	3.19	3.10
8	11.46	12.71	12.11	13.18	11.82	13.27	12.52	13.54	3.05	4.22	3.27	2.66
9	11.48	12.72	12.13	13.16	11.81	13.24	12.54	13.62	2.79	3.93	3.27	3.38
10	11.42	12.71	12.13	13.15	11.81	13.26	12.54	13.53	3.30	4.15	3.27	2.81
Avg.	11.42	12.73	12.10	13.15	11.84	13.26	12.53	13.59	3.52	3.95	3.38	3.21

The whole process of hybrid controller may be summarized as

- State location of start and target.
- Robot follows target until the obstacle is detected within the threshold range.
- Once an obstacle is detected, the MKH model activates.
- The initial inputs, DORO, DOLO, DOFO, are fed into the MKH model and find IPA as interim output.
- IPA along with DORO, DOLO, DOFO are fed to the FLC controller.
- Calculate FPA according to the rules of FLC.
- FPA is provided to the robots that help to move forward.

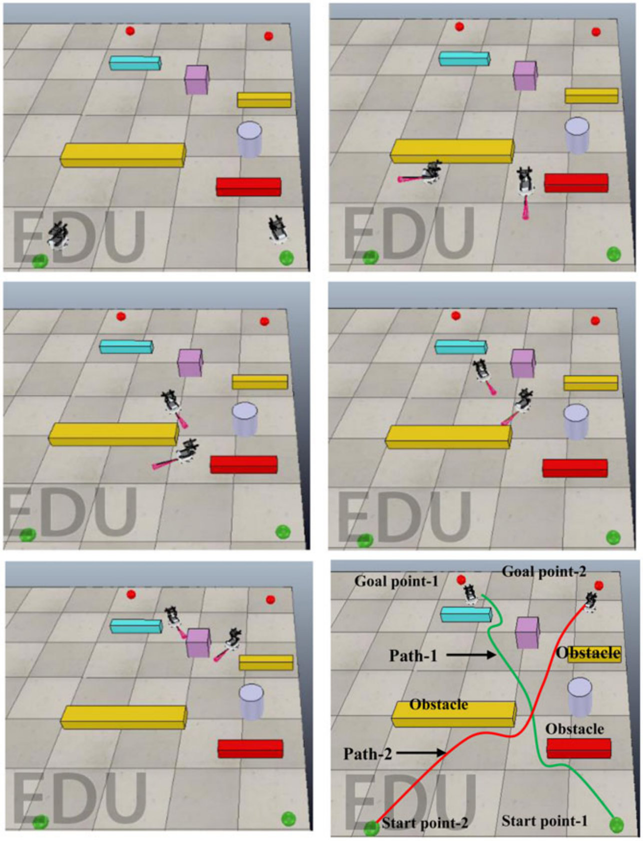

Figure 16. *V-Rep navigational analysis.*

5. Petri-net controller

The hybrid MKH-Fuzzy approach is intelligent enough to navigate from start to target points. However, it cannot perform well at conflict situation during multiple robot navigation due to several robots detecting multiple dynamic obstacles. At that time, one robot treats others as an obstacle, and confusion occurs among the robots regarding which robot should move first and complete the task. According to the controller, all the robots start their journey, and there is a chance of inter-collision. A Petri-net controller is added [7] to overcome the inter-collision situation and enhance the controller, whose function is to provide priority related to the motion of robots.

Figure 8 elaborates all stages of the Petri-net network model, and each phase are described as follows:

Stage 1: The first stage is the waiting stage for each randomly placed robot. Here, randomly implies robot location is unknown. It waits until the command is released to move towards the target.

Stage 2: In this stage, each robot starts its journey. They may sense some obstacles during navigation.

Stage 3: In this stage, robots find some obstacles.

Stage 4: This stage is termed as decision stage as the Petri-net controller decides the priority of movement. It implies the preference is given to that robot which is nearest to the goal-point among the robots. During the movement of priority robot, other robots act as stationary obstacles at their locations till the threshold range.

Stage 5: This stage is known as the scrutiny stage, where the robots search whether any conflict of movement exists or not. If not found any conflict, then move towards the target.

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

Table VII. Result of navigational analysis in V-Rep.

	Path length		Time consumption	
Sl. No.	R-1	R-2	R-1	R-2
1	220.45	223.65	17.02	17.29
2	221.41	223.72	17.14	17.27
3	220.64	223.41	17.20	17.26
4	220.52	222.42	17.16	17.28
5	221.63	223.67	17.11	17.34
6	221.75	223.48	17.09	17.26
7	221.74	223.58	17.14	17.26
8	221.58	223.47	17.15	17.27
9	221.61	223.57	17.42	17.21
10	221.77	223.71	17.31	17.42
Average	221.31	223.47	17.17	17.29

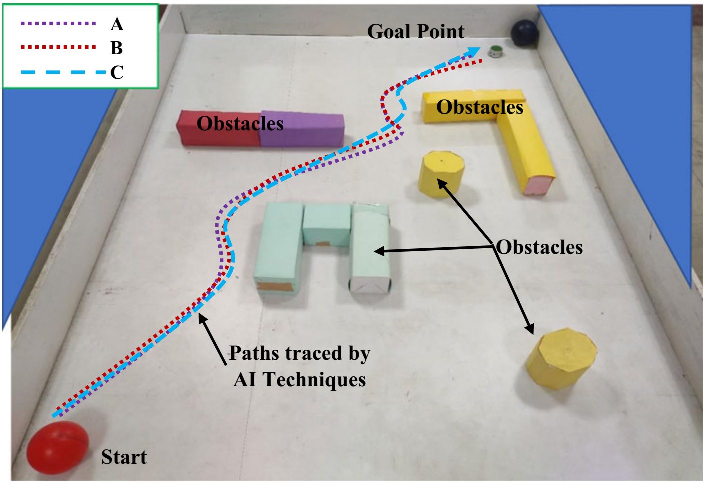

Figure 17. Comparison among PSO (Path-A), ABC (Path- B) and MKH-Fuzzy (Path-C) Techniques for path length and time consumption.

Stage 6: If the priority robot finds any new robot, it behaves like a stationary obstacle and waits until the priority robot crosses the threshold distance. Later, the waiting robot completes its task from stage 2.

With the above-discussed steps of the Petri-net controller, the navigation of multiple robots in a shared workspace can be performed with ease.

6. Execution of proposed hybrid MKH-fuzzy controller

The developed hybrid controller is implemented on simulation and real-time experiments by considering the Khepera-II robot (Fig. 9) on the navigation platform. Here, navigation of single and multiple robots has been performed in the designed environments. Only hybrid controller is implemented in single robot navigation; however, hybrid controller with Petri-net controller is executed in multiple robot analysis to avoid conflict situations.

Table VIII. *Comparative table for path length and time consumption.*

				% Improvement	
Sl. No.	**Algorithm/ Technique**	**Path length**	**Time consumption**	**In path length**	**In time consumption**
1	PSO	211.3	17.23	9.32	10.73
2	ABC	213.8	17.66	10.38	12.91
3	MKH-Fuzzy	191.6	15.38	Average = 9.85%	11.82%

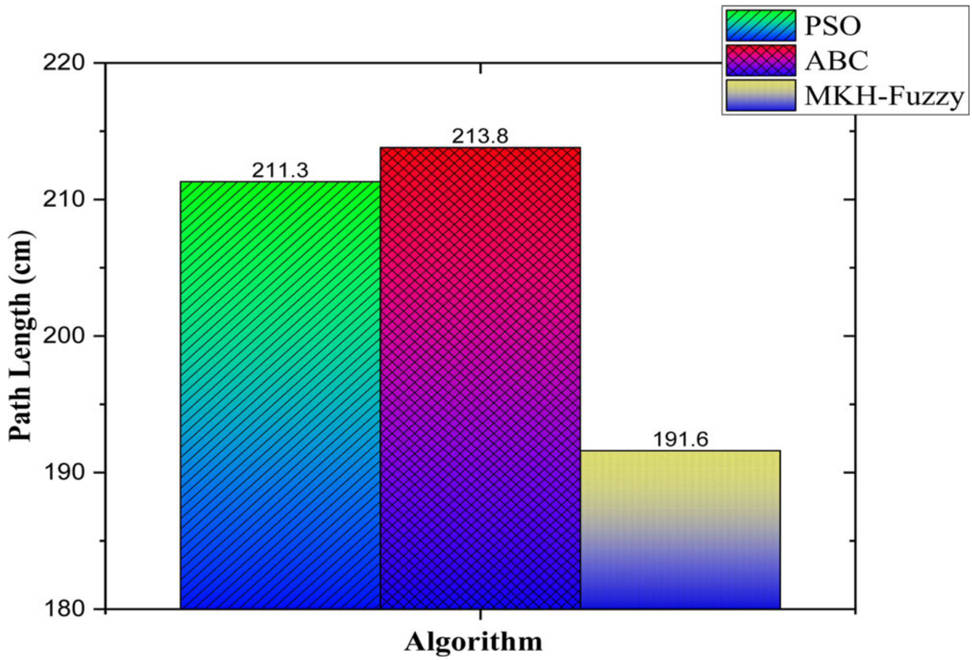

Figure 18. *Path length comparison plot.*

6.1. Navigation of single robot

As MATLAB has gained immense popularity among researchers to demonstrate navigational problems, it has been used as a simulation platform in single and multiple robot analysis. An arena of size $200 \times 200\, cm^2$ is allocated for analysis on both the simulation and experimental platforms. A cluttered environment has been created with the help of rectangular and hexagonal obstacles, as shown in experimental figures. The real-time experiment is conducted under laboratory conditions by creating the exactly same environment as simulation. The navigational analysis is shown in Figs. 10 and 11, and the results are recorded in Tables I and II.

After perusal of the results mentioned above, it has been observed that the hybrid controller performed well in both environments as earned less than 5% of deviation in results. In robotic research, it is said that below 5% is an acceptable range of deviation; actually, the reason behind this, is wheel slippage, friction, internet connectivity, etc.

6.2. Navigation of multiple robots

Multiple robot navigation is quite different from single robot navigation. As stated above, a Petri-net controller has been added for smooth negotiation of dynamic obstacles and avoidance of robots' inter-collision. The arena size is kept similar to single robot navigation; however, the workspaces have been

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

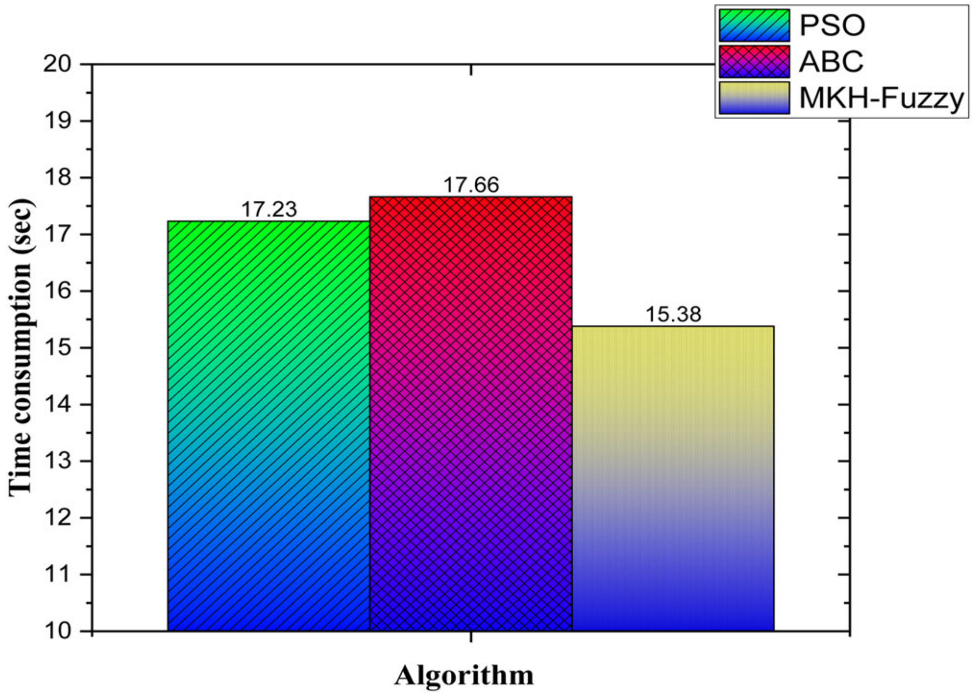

Figure 19. Time consumption comparison plot.

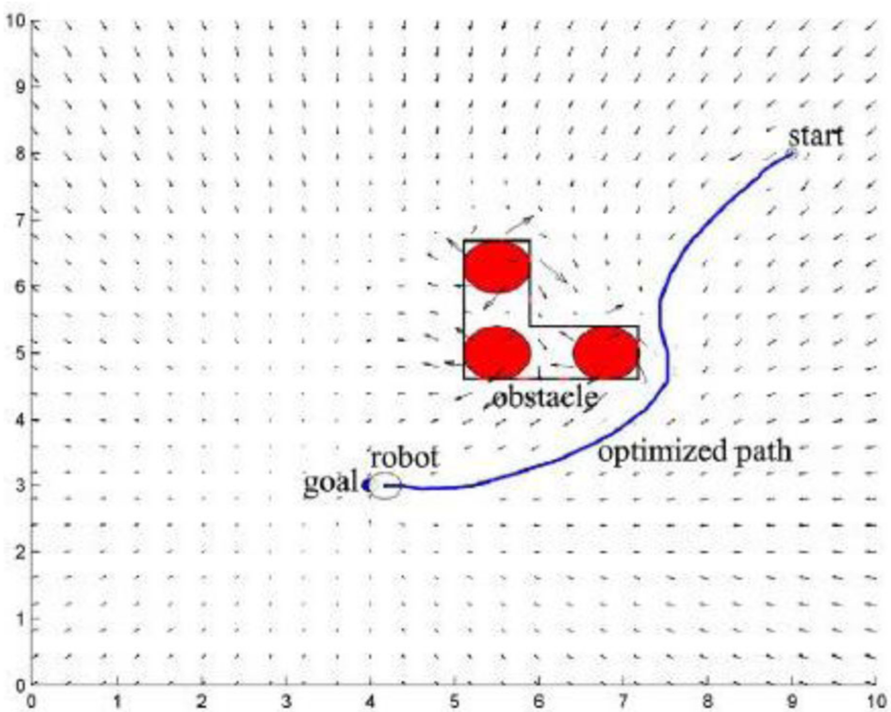

Figure 20. The path traced by bacterial potential field technique (PBF) [24].

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

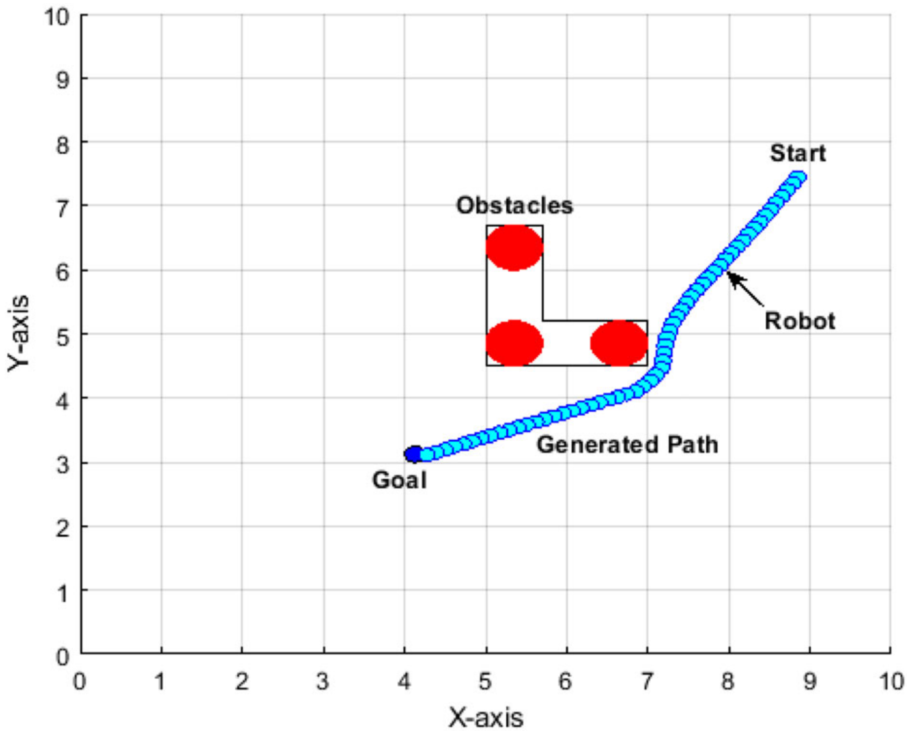

Figure 21. *The path traced by the MKH-Fuzzy controller (Proposed Technique).*

created by placing different blocks arbitrarily with predefined start points and target points. While applying the hybrid model with Petri-net controller, each robot starts moving towards its respective targets by avoiding static and dynamic obstacles. Moving robots are treated as dynamic obstacles in multiple robot analyses. Two robots and four robots have been used to execute the hybrid controller on MATLAB in scenes 1 and 2, as shown in Figs. 12 and 13. The outcomes of simulation analysis are validated through real-time experiments as shown in Figs. 14 and 15. The simulation and real-time experimental results are presented in Tables III, IV, V, and VI.

The evaluation of results from single and multiple robot navigation has imposed a satisfactory outcome that says that the developed hybrid controller is well established and performed in static and dynamic environments by optimal negotiation of obstacles. The hybrid controller has also revealed satisfactory time optimization. An acceptable range of deviation in results signifies proper execution and effective working of the proposed hybrid controller.

6.3. Analysis with Khepera-III robot

Along with a similar kind of robot, it is required to check the hybrid controller with a different one. Therefore, the Khepera-III robot is utilized in V-Rep simulation platform by implementing the developed controller. The outcomes of navigational analysis in V-REP has signified the compatibility of proposed controller with different platforms and dissimilar robots. The simulation analysis is shown in Fig. 16, and the results are in Table VII.

From the above Figures and Tables, it is observed that the proposed approach is adequately utilized the controller with Khepera-III robots. The deviation in the results shows good agreement between both the evaluating platforms as it is within 5%. The variation or error between both the platforms is occurred due to surface roughness, wheel slippage, surface friction, etc.

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

Table IX. *Comparison table for path length [24].*

Sl. No.	AI technique used	Path length (units)	Improvement (%)
1	GPF	8.338	16.53
2	PBFP	8.088	13.95
3	BPF	7.938	12.32
4	MKH-Fuzzy	6.96	**Average = 14.26%**

GPF: Genetic potential filed, PBFP: Pseudo-bacterial potential filed.

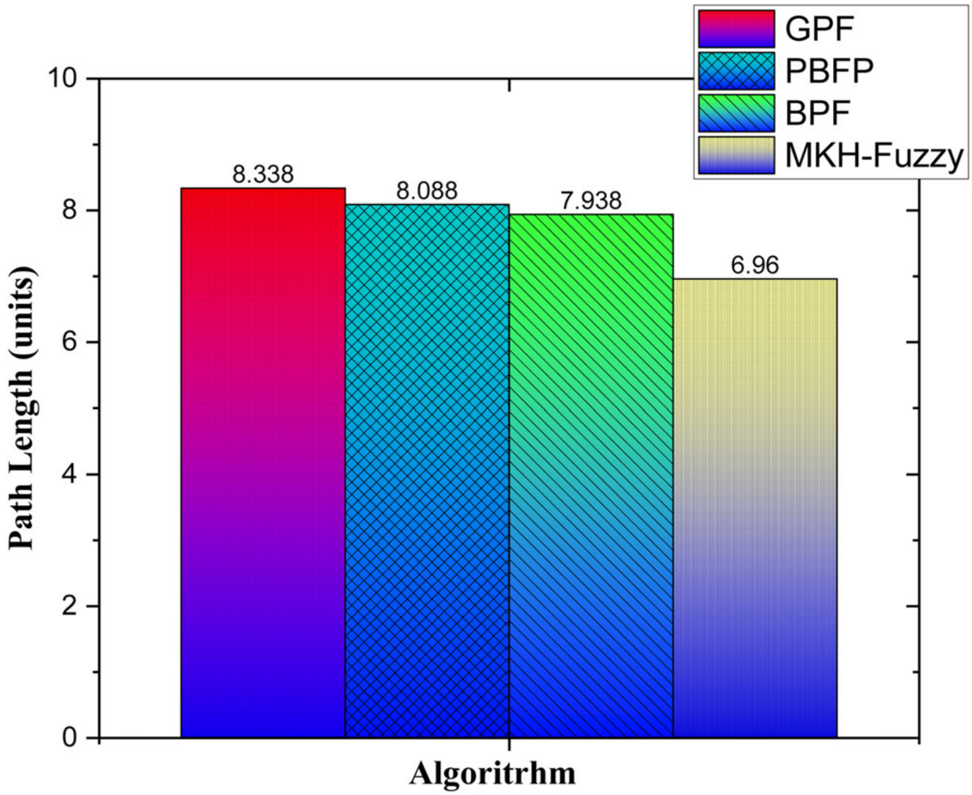

Figure 22. *Path length comparison histogram.*

7. Comparative analysis

In order to authenticate the simulation and real-time experimental results, a comparison between proposed controller and recognized or existing approach is required. Therefore, particle swarm optimization (PSO) and artificial bee colony (ABC) algorithms are considered for comparison in an environment. Figure 17 shows the trajectories traced by PSO, ABC, and MKH-Fuzzy techniques. The path lengths and time consumed by the robot using above mentioned techniques are recorded in Table VIII. Further, comparative bar charts are plotted as shown in Figs. 18 and 19, and convergence curves for the mentioned techniques are shown in Fig. 23. In addition to the above comparison, the proposed technique (MKH-Fuzzy) is again compared with the existing research paper [24], and an average improvement of 14.26% is found in path length. Figs. 20 and 21 show the paths generated by BPF [24] and proposed approaches, and the results are recorded in Table IX. Path length comparison bar chart is shown in Fig. 22.

The convergence graph shows the relation between path length obtained and number of iterations. Graph shows the fluctuation of length with iterations. Beginning of flat line indicates convergence

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

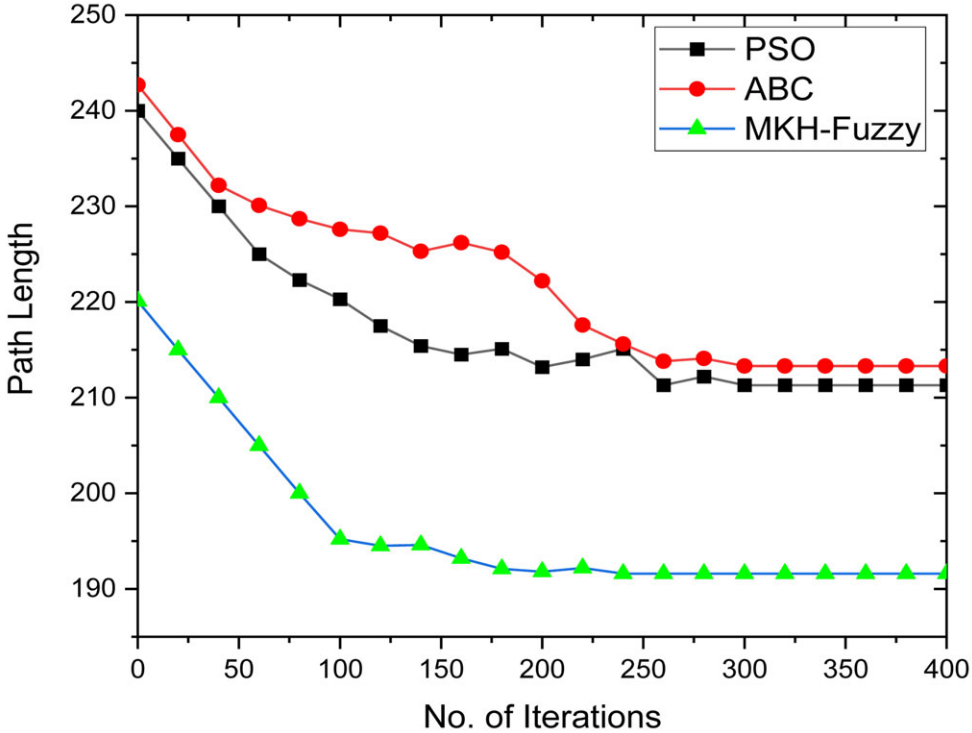

Figure 23. *Convergence curve between PSO, ABC, and MKH-Fuzzy Controller.*

of result implies the best solution is obtained. The graph also indicates the comparison between the approaches.

8. Conclusions and future scopes

This paper describes the navigational control of Khepera-II and Khepera-III robots using hybrid (MKH-Fuzzy) optimization approach aided Petri-net controller in unknown terrains. The aim of the proposed technique is achieved by successfully navigating the robots up to target without any collision within optimized time after avoiding local optima. The proposed approach is tested against other methods, and an average improvement of approximately 10% is remarked.

Additionally, the proposed technique is again tested against the existing research paper, and an average improvement of 14.26% is noted in terms of path length, which authenticates the proposed approach. In the future, the method may give an upper hand to the scholars of robotics to understand the scenario of route planning of robots. It can be applied to the real automation problem or in automatic robots. Besides, it may also be expanded by using this technique in dynamic environments.

Acknowledgement. The authors would like to thanks, National Institute of Technology, Rourkela, India for providing such a wonderful facility for research in the Robotics Laboratory.

Financial support. This research received no external funding.

Conflict of Interest statement. The authors do not have any conflict to disclose.

Author's contribution. Saroj Kumar – Conceptualization, Methodology, Design, Experiment, Drafting.
Dayal R. Parhi – Review, Editing, and Supervision.

https://doi.org/10.1017/S0263574722000698 Published online by Cambridge University Press

References

[1] A. H. Gandomi and A. H. Alavi, "Krill herd: A new bio-inspired optimization algorithm," *Commun. Nonlinear Sci. Numer. Simul.* **17**(12), 4831–4845 (2012).

[2] L. M. Abualigah, A. T. Khader, E. S. Hanandeh and A. H. Gandomi, "A novel hybridization strategy for krill herd algorithm applied to clustering techniques," *Appl. Soft Comput.* **60**, 423–435 (2017).

[3] D. C. Rao, M. R. Kabat, P. K. Das and P. K. Jena, "Cooperative navigation planning of multiple mobile robots using improved krill herd," *Arab. J. Sci. Eng.* **43**(12), 7869–7891 (2018).

[4] N. H. Singh and K. Thongam, "Mobile robot navigation using fuzzy logic in static environments," *Proc. Comput. Sci.* **125**, 11–17 (2018).

[5] C. H. Chen, S. Y. Jeng and C. J. Lin, "Mobile robot wall-following control using fuzzy logic controller with improved differential search and reinforcement learning," *Mathematics* **8**(8), 1254 (2020).

[6] C. Ben Jabeur and H. Seddik, "Design of a PID optimized neural networks and PD fuzzy logic controllers for a two-wheeled mobile robot," *Asian J. Control* **23**(1), 23–41 (2021).

[7] M. K. Muni, D. R. Parhi, P. B. Kumar and S. Kumar, "Motion control of multiple humanoids using a hybridized prim's algorithm-fuzzy controller," *Soft Comput.* **25**, 1159–1180 (2020). https://doi.org/10.1007/s00500-020-05212-z.

[8] M. K. Muni, D. R. Parhi, P. B. Kumar, C. Sahu, P. R. Dhal and S. Kumar, "Global Path Optimization of Humanoid NAO in Static Environment Using Prim's Algorithm," **In:** *Intelligent Systems* (Springer, Singapore, 2021) pp. 25–34. https://doi.org/10.1007/978-981-33-6081-5_3.

[9] S. Kumar, K. K. Pandey, M. K. Muni and D. R. Parhi, "Path Planning of the Mobile Robot Using Fuzzified Advanced Ant Colony Optimization," **In:** *Innovative Product Design and Intelligent Manufacturing Systems* (Springer, Singapore, 2020) pp. 1043–1052. https://doi.org/10.1007/978-981-15-2696-1_101.

[10] S. Kumar, D. R. Parhi, A. K. Kashyap and M. K. Muni, "Static and dynamic path optimization of multiple mobile robot using hybridized fuzzy logic-whale optimization algorithm," *Proc. Inst. Mech. Eng. C J. Mech. Eng. Sci.* **235**(21), 5718–5735 (2021). https://doi.org/10.1177/0954406220982641.

[11] P. K. Mohanty and D. R. Parhi, "Optimal path planning for a mobile robot using cuckoo search algorithm," *J. Exp. Theor. Artif. Intell.* **28**(1–2), 35–52 (2016).

[12] B. K. Patle, D. R. K. Parhi, A. Jagadeesh and S. K. Kashyap, "Matrix-binary codes based genetic algorithm for path planning of mobile robot," *Comput. Electr. Eng.* **67**, 708–728 (2018). https://doi.org/10.1016/j.compeleceng.2017.12.011.

[13] S. Kumar, D. R. Parhi, M. K. Muni and K. K. Pandey, "Optimal path search and control of mobile robot using hybridized sine-cosine algorithm and ant colony optimization technique," *Ind. Robot* **47**(4), 535–545 (2020). https://doi.org/10.1108/IR-12-2019-0248.

[14] M. K. Singh and D. R. Parhi, "Path optimisation of a mobile robot using an artificial neural network controller," *Int. J. Syst. Sci.* **42**(1), 107–120 (2011). https://doi.org/10.1080/00207720903470155.

[15] Parhi, D. R., Pradhan, S. K., Panda, A. K. and Behera, R. K., "The stable and precise motion control for multiple mobile robots," *Appl. Soft Comput.*, **9**(2), 477–487 (2009). ISSN: 1568–4946. https://doi.org/10.1016/j.asoc.2008.04.017.

[16] A. Pandey, S. Kumar, K. K. Pandey and D. R. Parhi, "Mobile robot navigation in unknown static environments using ANFIS controller," *Perspect. Sci.*, **8**, 421–423 (2016). https://doi.org/10.1016/j.pisc.2016.04.094.

[17] A. Pandey and D. R. Parhi, "MATLAB simulation for mobile robot navigation with hurdles in cluttered environment using minimum rule based fuzzy logic controller," *Proc. Technol.*, **14**(1), 28–34 (2014).

[18] P. K. Mohanty and D. R. Parhi, "A New Real Time Path Planning for Mobile Robot Navigation using Invasive Weed Optimization Algorithm," **In:** *Gas Turbine India Conference* (Vol. **49644**, American Society of Mechanical Engineers, 2014) p. V001T07A002.

[19] K. M. Sim and W. H. Sun, "Ant colony optimization for routing and load-balancing: survey and new directions," *IEEE Trans. Syst. Man Cybern. A Syst. Hum.* **33**(5), 560–572 (2003).

[20] D. T. Pham and D. R. Parhi, "Navigation of multiple mobile robots using a neural network and a Petri Net model," *Robotica* **21**(1), 79–93 (2003).

[21] J. K. Pothal and D. R. Parhi, "Navigation of multiple mobile robots in a highly clutter terrains using adaptive neuro-fuzzy inference system," *Rob. Auton. Syst.* **72**, 48–58 (2015).

[22] S. Kundu and D. R. Parhi, "Navigation of underwater robot based on dynamically adaptive harmony search algorithm," *Memetic Comput.* **8**(2), 125–146 (2016).

[23] M. A. Contreras-Cruz, V. Ayala-Ramirez and U. H. Hernandez-Belmonte, "Mobile robot path planning using artificial bee colony and evolutionary programming," *Appl. Soft Comput.* **30**, 319–328 (2015).

[24] L. Fen, Z. Jiang-hai, S. Xiao-bo, Z. Pei-ying, F. Shi-hui and L. Zhong-jie, "Path Planning of 6-DOF Humanoid Manipulator based on Improved Ant Colony Algorithm," **In:** *2012 24th Chinese Control and Decision Conference (CCDC)* (IEEE, 2012) pp. 4158–4161.

[25] M. K. Muni, D. R. Parhi, P. Kumar, K. K. Pandey, S. Kumar and A. Chhotray, Sugeno Fuzzy Logic Analysis: Navigation of Multiple Humanoids in Complex Environments," **In:** *International Conference on Artificial Intelligence in Manufacturing & Renewable Energy (ICAIMRE)* (2019).

[26] M. K. Muni, D. R. Parhi, P. B. Kumar, C. Sahu and S. Kumar, "Towards motion planning of humanoids using a fuzzy embedded neural network approach," *Appl. Soft Comput.* **119**, 108588 (2022).

[27] M. K. Muni, S. Kumar, D. R. Parhi and K. K. Pandey, "Water cycle algorithm: an approach for improvement of navigational strategy of multiple humanoid robots," *Robotica*, 1–19 (2021). https://doi.org/10.1017/S0263574721000837.

[28] S. Kumar, D. R. Parhi, K. K. Pandey and M. K. Muni, "Hybrid IWD-GA: an approach for path optimization and control of multiple mobile robot in obscure static and dynamic environments," *Robotica*, 1–28 (2021). https://doi.org/10.1017/S0263574721000114.

[29] O. Montiel, U. Orozco-Rosas and R. Sepúlveda, "Path planning for mobile robots using Bacterial Potential Field for avoiding static and dynamic obstacles," *Expert Syst. Appl.* **42**(12), 5177–5191 (2015). https://doi.org/10.1016/j.eswa.2015.02.033.

[30] A. Ahmed, A. Maged, A. Soliman, H. El-Hussieny and M. Magdy, "Space deformation based path planning for mobile robots," *ISA Trans.* (2021). https://doi.org/10.1016/j.isatra.2021.08.019.

[31] A. L. A. Bolaji, M. A. Al-Betar, M. A. Awadallah, A. T. Khader, & L. M. Abualigah (2016). A comprehensive review: Krill Herd algorithm (KH) and its applications. *Appl. Soft Comput.*, **49**, 437–446.

[32] S. Kumar, M. K. Muni, K. K. Pandey, A. Chhotray and D. R. Parhi, "Path Planning and Control of Mobile Robots using Modified Tabu Search Algorithm in Complex Environment," **In:** *International Conference on Artificial Intelligence in Manufacturing & Renewable Energy (ICAIMRE)* (2019). https://doi.org/10.2139/ssrn.3539922.

[33] S. Kumar, D. R. Parhi, A. K. Kashyap, M. K. Muni and P. R. Dhal, "Navigational Control and Path Optimization of Mobile Robot Using Updated Sine–Cosine Algorithm in Obscure Environment," **In:** *Current Advances in Mechanical Engineering* (Springer, Singapore, 2021) pp. 989–996. https://doi.org/10.1007/978-981-33-4795-3_91.

[34] D. Wang, S. Chen, Y. Zhang and L. Liu, Path planning of mobile robot in dynamic environment: Fuzzy artificial potential field and extensible neural network. *Artif. Life Robot.*, **26**(1), 129–139 (2021). https://doi.org/10.1007/s10015-020-00630-6.

[35] M. Dirik, A. F. Kocamaz and O. Castillo, "Global path planning and path-following for wheeled mobile robot using a novel control structure based on a vision sensor," *Int. J. Fuzzy Syst.* **22**(6), 1880–1891 (2020). https://doi.org/10.1007/s40815-020-00888-9.

[36] Q. Luo, H. Wang, Y. Zheng and J. He, "Research on path planning of mobile robot based on improved ant colony algorithm," *Neural Comput. Appl.* **32**(6), 1555–1566 (2020). https://doi.org/10.1007/s00521-019-04172-2.

[37] J. Li, J. Wang, H. Peng, Y. Hu and H. Su, "Fuzzy-torque approximation-enhanced sliding mode control for lateral stability of mobile robot," *IEEE Trans. Syst. Man Cybernet. Syst.* **52**(4), 2491–2500 (2022). https://doi.org/10.1109/TSMC.2021.3050616.

[38] T. A. Teli and M. A. Wani, "A fuzzy based local minima avoidance path planning in autonomous robots," *Int. J. Inf. Technol.* **13**(1), 33–40 (2021). https://doi.org/10.1007/s41870-020-00547-0.

[39] P. G. Luan and N. T. Thinh, "Hybrid genetic algorithm based smooth global-path planning for a mobile robot," Mech. Des. Struct. Mach., 1–17 (2021). https://doi.org/10.1080/15397734.2021.1876569.

[40] P. Kim, J. Park, Y. K. Cho and J. Kang, "UAV-assisted autonomous mobile robot navigation for as-is 3D data collection and registration in cluttered environments," *Autom. Constr.* **106**, 102918 (2019). https://doi.org/10.1016/j.autcon.2019.102918.

[41] H. Hu, K. Zhang, A. H. Tan, M. Ruan, C. Agia and G. Nejat, "A sim-to-real pipeline for deep reinforcement learning for autonomous robot navigation in cluttered rough terrain," *IEEE Rob. Autom. Lett.*, **6**(4), 6569–6576 (2021). https://doi.org/10.1109/LRA.2021.3093551.

Cite this article: S. Kumar and D. R. Parhi (2022). "Trajectory planning and control of multiple mobile robot using hybrid MKH-fuzzy logic controller", Robotica **40**, 3952–3975. https://doi.org/10.1017/S0263574722000698