Robotica (2025), **43**, pp. 2273–2303
doi:10.1017/S0263574725000712

RESEARCH ARTICLE

An improved butterfly optimization algorithm-based path navigation of humanoid robots in an unfamiliar setting

Himansu Sekhar Dash[1,2], Dayal R. Parhi[1], Manoj Kumar Muni[3] and Pinaki Das[1]

[1]Robotics Laboratory, Department of Mechanical Engineering, National Institute of Technology, Rourkela, Odisha, India
[2]Department of Production Engineering, Indira Gandhi Institute of Technology, Sarang, Odisha, India
[3]Department of Mechanical Engineering, Indira Gandhi Institute of Technology, Sarang, Odisha, India
Corresponding author: Himansu Sekhar Dash; Email: 919me5062@nitrkl.ac.in

Received: 25 November 2024; **Revised:** 4 February 2025; **Accepted:** 15 April 2025; **First published online:** 4 June 2025

Keywords: humanoid robots; navigation; unknown environment; butterfly optimization algorithm; global optimization; Webots; NAO

Abstract

The path navigation of robot in an entirely known space is presented by various researchers in the recent times. The navigational complexity arises when a robot moves in a completely unknown and complex environment from one defined start to a designated desired location. As the success of the nature-inspired algorithms in the unclear navigational problem is better, therefore, an improved butterfly optimization algorithm (IBOA) to determine the optimal feasible path for a humanoid robot navigating through a platform cluttered with both known and unfamiliar barriers is presented in this study. The BOA is inspired by the food-gathering habits of butterflies, where the sense of smell is the vital parameter in the global optimal search. However, the performance of this technique in the complex environment is poor, as a result, the chances of being trapped in local minima are more. Hence, the BOA is improved by using a nonlinear weight reduction strategy in updating the position of the butterflies in every iteration. The simulation is carried out in the Webots platform by considering variable-legged robot, NAO, in an unfamiliar environment. The outcomes derived from the simulation and real assessments demonstrate the potential of the proposed technique and compare with other existing algorithms, which highlights the potential and efficacy of the proposed IBOA algorithm.

1. Introduction

Identifying the proper route of a manipulator in a real working environment is still a difficult task for many researchers, especially when you are working with an autonomous legged robot. In this paper, the humanoid NAO path navigation is given special attention by using an improved BOA (IBOA) for finding out the optimal feasible path. NAO H25, V4.0, having 25 degrees of freedom and developed by Aldebaran Robotics [1], is taken into consideration. NAO has 25 degrees of freedom with various sensors attached to it, such as touch sensors, ultrasonic sensors, tactile sensors, feet bumpers, magnetic rotary encoders, joint position sensors, etc. NAO can be customized to handle numerous tasks using various programing languages like Python, MATLAB, C++, and Choregraphe, a graphical programming interface. The specification of NAO is presented in Table I. Special features such as localization, perception and cognition, human-robot interaction, locomotion, and choreographer features make this robot different from other autonomous robots. It supports multiple operating systems and platforms, making it flexible for different research and development needs. The route optimization of a robot refers to identification of an obstacle-free path and reaching the goal point based on an optimization criterion in a given work environment. The optimization criteria used is such as route distance, turning angle, route safety, and route time. Various techniques are being used in the route optimization of the robot, and they are broadly categorized into classical and heuristic types. Literature reviews of such techniques used for path navigation problem are presented in the following section.

Table I. *Specifications of the humanoid NAO [1].*

Sl. No.	Item	Specification
1	Version	NAO H25 V4.0
2	Developed by	Aldebaran Robotics
3	Dimension, Material	574 H x 275W x 311D (mm), ABS-PC/PA-66/XCF-30
4	Weight	5.4 KG
5	Language	Understand and revert seven different languages
6	Cameras	MT9M114, SOC Image Sensor, 1280x960 resolution at 30 FPS.
7	Motherboard	ATOM Z530 1.6 GHz CPU 1 GB RAM, 2 GB Flash memory, 8 GB Micro SDHC
8	Input	100 to 240 V AC, 50-60 Hz and maximum 1.2A
9	Battery	Lithium-Ion, 21.6 V, 2014AH, 27.6 WH . Working: up to 2 hour.
10	DOF	25 DOF, Head: 2 DOF, Arm: 6 DOF, Leg: 5 DOF. Hip Joint: 1 DOF.

Numerous optimization strategies have been developed for mobile robot path navigation by many researchers; however, their applicability is limited when considering humanoid robots. Literature reviews of some of them are present here. The idea of a fuzzy system is proposed by Singh et. al. [2] for route optimization of wheeled robot in a known platform. To determine the effectiveness of the suggested concept, the authors have conducted simulations and experiments in a variety of terrains. Yuxiong et.al. [3] presented a three-dimensional potential field approach in path navigation of an autonomous vehicle. Weihao et. al. [4] used the potential field techniques for identification of path of an omnidirectional mobile robot. The fuzzy technique presented by Lee et.al [5] is used to prioritize the head direction of the manipulator to find out a barrier-free route in a cluttered environment. The kinematic and dynamic models of the wheeled manipulator used in the research serve as the foundation for the suggested approach. Improved velocity potential field algorithm is used by Xia et. al. [6] for the trajectory planning of robot arms in the medical surgical operation. Chen et. al [7] focused on implementation of the probabilistic roadmap method in route optimization for manipulators. Rapidly exploring random tree (RRT) method is used by Mthabela et. al [8] for route optimization of the wheeled robot. The suggested approach significantly reduces the amount of time needed to compute the feasible path. Saeed et al. [9] focused on a Boundary Node Method for identifying the initial feasible path during path optimization of wheeled robot. Cai et. al. [10] presented human-robot interaction in construction that is both safe and effective, incorporating the anticipated movements of construction workers into robot route optimization based on Deep Reinforcement Learning (DRL). Yang et. al. [11] focused on the Q-network (DQN) method is a DRL system that solves the multi-robot route planning problem by combining the Q-learning algorithm. Zhu et. al. [12] demonstrated the use of the Simulated Annealing Algorithm and the Artificial Potential Field technique for global route finding of a wheeled robot. Low et. al. [13] outlined a modified Q-learning method for route identification of a three-wheeled robot. The nature-inspired model such as Particle Swarm Optimization (PSO) [14], Ant Colony Optimization (ACO) [15], Genetic Algorithm (GA) [16], Neural Network (NN) [17], Firefly algorithm (FA) [18], Grey Wolf Optimization (GWO) [19], Bacterial Forging Optimization [20], Cuckoo Search (CS) [21], and many more developed by authors in the route navigation of the manipulator in a cluttered region are discussed in this section. Li Lu et. al. [22] focused on the PSO technique in path navigation of mobile robot. First, the robot's path finding problem is converted into a minimization model, and then a particle fitness value is defined according to the target location and obstacle position within the working environment, and identifies the global best of the manipulator. The outcomes demonstrate that the suggested methods are workable substitutes for resolving the robot route finding issue in the presence of hazardous risk sources.

The Hybridization techniques, as developed by the authors, are showing better results in the path planning problem; some of them are discussed here. The particle filter and PSO model inspired hybrid localization technique is presented by Zhang et. al. [23]. Das et. al. [24] suggested a unique method for hybridizing the differentially perturbed velocity (DV) algorithm with improved PSO (IPSO) to identify the best travel trajectory for several manipulators in a cluttered environment, and for the best trajectory path length and arrival time, the suggested IPSO-DV outperforms IPSO and DE. Purian and Sadeghian [25] demonstrate hybridize technique by using ACO and Fuzzy model for the route optimization of wheeled manipulator in an uncertain space. In this approach, the variables of the fuzzy ruler are optimized by the ACO during path optimization in complex space, and the optimum route is identified by the fuzzy logic. Nie and Zhao [26] proposed a Dijkstra-ACO hybridized technique for route finding of robot. Initially, the path to reach the goal is chosen by the Dijkstra first, and then the path optimization will be done by the ACO in the proposed technique. The feasibility of the said algorithm is presented in a MATLAB simulation environment.

By utilizing the fuzzy information such as fuzzy theory and fuzzy interference function in the Neural network optimization method Wang et al. [27] proposed a hybridized technique such as Fuzzy NN in the route optimization of robot. The theory of NN and GA is combined by the Noguchi and Terao [28] in the trajectory planning of the agriculture wheeled manipulator. The non-linearity in the motion planning is primarily focused on by the author with NN-GA hybrid technique. A navigational hybrid model focused on NN and Hierarchical Reinforcement Learning is addressed by Yu et al. [29]. The author was able to reduce the route steps, shorten the path planning time, and enhance the smoothness of the planned path by employing this technique. To enhance the efficacy of the model and reduce the cost-performance metric in route optimization, a hybrid method based on PSO, FA, and CS is suggested by the author, Garip et al. [30]. The author compared the virtual and real experimental results derived from the hybrid CS-PSO-FA with the results derived from CS, PSO, and FA in a similar environment. To achieve equilibrium between exploration and exploitation and to avoid pre-mature convergence Panda et al. [31] have combined an FA with a heuristics population-based Invasive Weed Optimization model and proposed a hybrid IWFO technique. Mohanty and Parhi [32] demonstrated a hybrid CS-ANFIS model by combining the CS and adaptive neuro-fuzzy interference system (ANFIS). The CS approach is taken in consideration to train the premise portion of this new hybrid navigational methodology, while the least squares estimation approach is used to prepare the subsequent parameters of the ANFIS. Yu et. al. [33] focused on a hybrid technique by combining GWO and differential evolution for route optimization of UAV. Sahoo et. al. [34] chose the GWO technique with the GA and modeled a hybrid GWO for route finding of autonomous underwater vehicle.

The wide literature suggests that path planning using various techniques is a very hot topic. Many researchers have used various techniques to optimize it for identifying the shortest route, steering angle, and safety distance from the obstacle during navigation of the robot. The navigational analysis in autonomous robots is largely found in navigation of the mobile robots compared to humanoid or legged robots, and also many metaheuristic algorithms developed by different authors have not been tested in the route optimization of legged robots. Therefore, the current study is giving emphasis on implementation of improved version of butterfly optimization algorithm (IBOA) in order to effectively navigate humanoid NAO. The efficacy of the BOA algorithm over others and the improvement in the BOA is highlighted in the following section. Single and multiple NAOs implement the described algorithm in simulation and experimental platforms, with observed satisfying outcomes.

2. Overview of proposed optimization strategy

The nature-inspired optimization techniques are used by various authors for route optimization of robot in unclear terrain. From the literature, it is found that many optimization techniques have been implemented by various authors for the path navigation. The BOA is a type of probabilistic search technique that is modeled after the resource-seeking behavior of butterflies. The advantages over other optimization techniques, such as avoiding the local minima and fast convergence rate, with various developments in the technique, are briefly discussed in this section.

2.1. Butterfly Optimization Algorithm (BOA)

This optimization technique solves the global optimization problems by imitating the food finding as well as mating characteristics of butterflies. The location of nectar or mating partner is basically identified by

Table II. *Representation of butterfly optimization algorithm parameters.*

Sl. No.	Symbol	Representation
1	Z_i^t	Solution Space for Z_i, where i represents of butterfly strength and t shows iteration number
2	G^*	Present best solution
3	f_i	Fragrance of the i^{th} butterfly
4	R	Randomness [0,1]
5	Z_j^t, Z_k^t	Here j & k represent the j^{th} and k^{th} butterfly
6	β	Switch Probability

the sense of smell of butterflies. The fitness of a butterfly will vary according to its fragrance intensity, and it varies from location to location. This fragrance will play an important role in creating a social knowledge network among butterflies. The global search occurs mainly when a butterfly finds and senses the fragrance of abother butterfly and moves towards it. Whereas local search is the phase when it is unable to identify and sense any fragrance from others, then it moves randomly to somewhere else; this is known as local search [35].

Three essential elements form the basis of the overall sensing and processing approach used as the sensory modality (c), stimulus intensity (I), and power exponent (a). A key parameter in the BOA is the fluctuation of I and f's formulation. I is related to the fitness function. f is relative, though, so other butterflies ought to be able to detect it. c is utilized, in accordance with Steven's power law (Stevens 1975), to differentiate smell from other modalities. Now, f increases more quickly than I when the butterfly with fewer I approaches the butterfly with more I. In order to obtain this level of absorption, the power exponent parameter a is used to vary f. Utilizing these ideas, the smell in BOA is derived as function of stimulus intensity, which is mentioned below in Eq. (1).

$$f = cI^a \tag{1}$$

The value of a and c is taken in the range between 0 and 1. The value $a = 1$ represents zero absorption of smell, i.e., the smell emitted by a single butterfly is received in identical manner by other butterflies. Whereas the vale $a = 0$ shows the emitted fragrance cannot be sensed by other butterflies. The convergence of the BOA model primarily depends upon the value of a and c. The BOA is performed in three phases such as Initialization, Iteration, and Final phase. Every BOA run begins with the initialization phase, followed by an iterative search, and a final phase where the algorithm is eventually stopped after the global best result is shown. The fitness function and its searching space are defined by the algorithm at the initialization stage. Additionally, values are assigned to the variables used in BOA. The method then generates an initial group of butterflies for optimization after the values have been set. Then, by evaluating the fitness value and fragrance, the locations of the butterflies are created within the searching space. The algorithm initialization phase stops, and iteration phase starts with the artificial butterflies that are created in the initialization phase. The parameters of the BOA are depicted in Table II.

In the second phase, which is the iteration phase, the model runs through a number of iterations. Every butterfly in the solution space moves to a new location after each iteration, after which their fitness values are assessed. The algorithm begins by determining each butterfly's fitness value at each of its various locations throughout the solution space. The smell of the butterflies is created by using Eq. (1). The iteration phase is based on the two phases, such as global and local search phase. The movement of the butterfly to the best butterfly (G^*) in the solution space is termed as global search and it is shown in Eq. (2). The local random movement is known as local search when the butterflies is not able to sense the fragrance of best butterflies it will take a random walk by using Eq. (3). The Eq. (3) represents here the local search of the BOA. The equation of universal and local search, as derived by the author [35], is mentioned in the Eqs. 3 and 4, respectively.

$$Z_i^{t+1} = Z_i^t + (R^2 \times G^* - Z_i^t) \times f_i \tag{2}$$

$$Z_i^{t+1} = Z_i^t + (R^2 \times Z_j^t - Z_k^t) \times f_i \tag{3}$$

These two local search and global search can be switched by a parameter β, which represents the switch probability. The value of stimulus intensity (c) greatly influences the convergence of the algorithm. The value of c should not be too big or too small, and to avoid the premature convergence, the author [35] demonstrates the calculation of c as present in Eq. (4).

$$c^{i+1} = c^i + \left(\frac{0.025}{c^i \times Max\_it} \right) \tag{4}$$

Where i and Max_it denote the current iteration and maximum iteration of the algorithm, respectively. The iteration phases continue until the stopping criteria are fulfilled. The location of the manipulator, such as the manipulator reaching to the desired point, is defined as the stopping criteria in this path navigational problem. The details of the three phases are mentioned below in the flowchart of BOA and pseudocode of BOA.

2.2. Proposed improved BOA (IBOA)

It has been found from the literature that the possibility of capture in the local minima is high in the conventional BOA technique. Consequently, this study introduces an improved BOA (IBOA) technique by using the Levy flight, and a nonlinear strategy of decreasing inertia weight factor is implemented in the basic BOA architecture. The Levy flight [36] is one operator that is used for short range through local search and occasional movements for long-distance search. As the butterflies are searching for the food in a very unpredictable region so the Levy flight will improve the efficacy of the searching and also the diversity among the butterflies. While the algorithm is not able to detect any nearby solution, then the butterfly position will be updated by the operator by using Eq. (5).

$$Z_i^{t+1} = Z_i^t + (L\hat{e}vy\ (D)\ Z_j^t - Z_k^t) \times f_i \tag{5}$$

Here, the current iteration is denoted by t, and position vector dimension is denoted by D.

$$L\hat{e}vy\ (z) = 0.01 \times \frac{R_1 \times \rho}{|R_2|^{\frac{1}{\alpha}}} \tag{6}$$

Here, R1 and R2 denote two random number in [0,1], α is a constant ($\alpha = 1.5$ here) and ρ calculated as mentioned below in Eq. (7).

$$\rho = \left(\frac{\Gamma(1+\alpha) \times \sin\frac{\pi\alpha}{2}}{\Gamma\left(\frac{1+\alpha}{2}\right) \times \alpha \times 2^{\left(\frac{\alpha-1}{2}\right)}} \right)^{\frac{1}{\alpha}} \tag{7}$$

In BOA, the search agents are the butterflies. All the butterflies are searching for the best butterfly, which is having a large fragrance value. If the best butterfly is trapped in the local optimal, then the other butterflies will also be trapped. Therefore, to overcome this, in this paper, a nonlinear decreasing inertia weight [37] has been introduced during the global search phase. The weights are updated as shown below:

$$W = W_{min} \left(\frac{W_{max}}{W_{min}} \right)^{\left(\frac{1+\vartheta t}{Max\_it}\right)} \tag{8}$$

The butterflies positions will be updated based on the nonlinear decreasing inertia weight, as per Eq. (9).

$$Z_i^{t+1} = W \times Z_i^t + (R^2 \times G^* - Z_i^t) \times f_i \tag{9}$$

Here, W_{min} and W_{max} represent the minimum and maximum inertia weights of 0.4 and 0.95, respectively. The acceleration factor is represented by ϑ with a value 10.

The pseudocode of the IBOA is presented below.

```
Objective Function $f(Z)$, $Z = (Z_1, Z_2, \ldots\ldots Z_d)$, $d =$ Dimensions
Generation of the strength of $M$ butterflies $Z_i^t$ for $i = 1, 2, 3 \ldots M$
The fitness value of $f(Z_i)$ which shows stimulus intensity of $I_i$ at $Z_i$
Substitute parameter value of $c$, $a$ and $\beta$
while max iteration not reached
        For each butterfly bf in population do
            Calculate fragrance for bf using Eq. 1
        end for
        find the best bf
        for each butterfly bf in population do
            Generate random number $R \in [0,1]$
                if $R < \beta$ then
                Move towards best butterfly using Eq. 9
                else
                Move randomly using Eq. 5
                end if
        end for
        Update value of $c$ using Eq. 4
end while
Output the optimum solution obtained.
```

The flowchart of the suggested IBOA is shown below in Figure 1.

2.3. Mathematical model of path navigation

The workspace modeling, path searching, and path smoothness are the three key components of the robot navigational problem. In a real-world setting, a robot must navigate complex and rapidly changing conditions. To standardize the experimental environment, the WEBOT simulation platform was employed for modeling the environment. The robot route finding problem is then converted into using algorithms to guide the robot from the starting position to the desired location, avoiding obstacles along the way and securing an optimal path that is free from collisions. Based on environmental modeling, the challenge of identifying an optimal path is transformed into the task of optimizing a fitness function to achieve the best path. Considering an obstacle-free path, the distance between the start and to goal is formulated as an objective function as shown in Eq. (10).

$$F_D = \sqrt{\left(X_{st} - X_{go}\right)^2 - \left(Y_{st} - X_{go}\right)^2} \tag{10}$$

where F_D represents the shortest length from start and goal point. (X_{st}, Y_{st}) and (X_{go}, Y_{go}) represent the start and goal point, respectively. However, in real working environments, the robot will face obstacles that hinder its direct forward movement. The workspace model of the manipulator is assigned with a Cartesian reference system with some nodes as $Z_i = (X_i, Y_i), i = 1, 2, 3, 4 \ldots n$. The total length between the search agents formed by the model is shown in Eq. (11).

$$D = \sum_{k=1}^{m} \sqrt{(X_{k+1} - X_k)^2 + (Y_{k+1} - Y_k)^2}, (m < n) \tag{11}$$

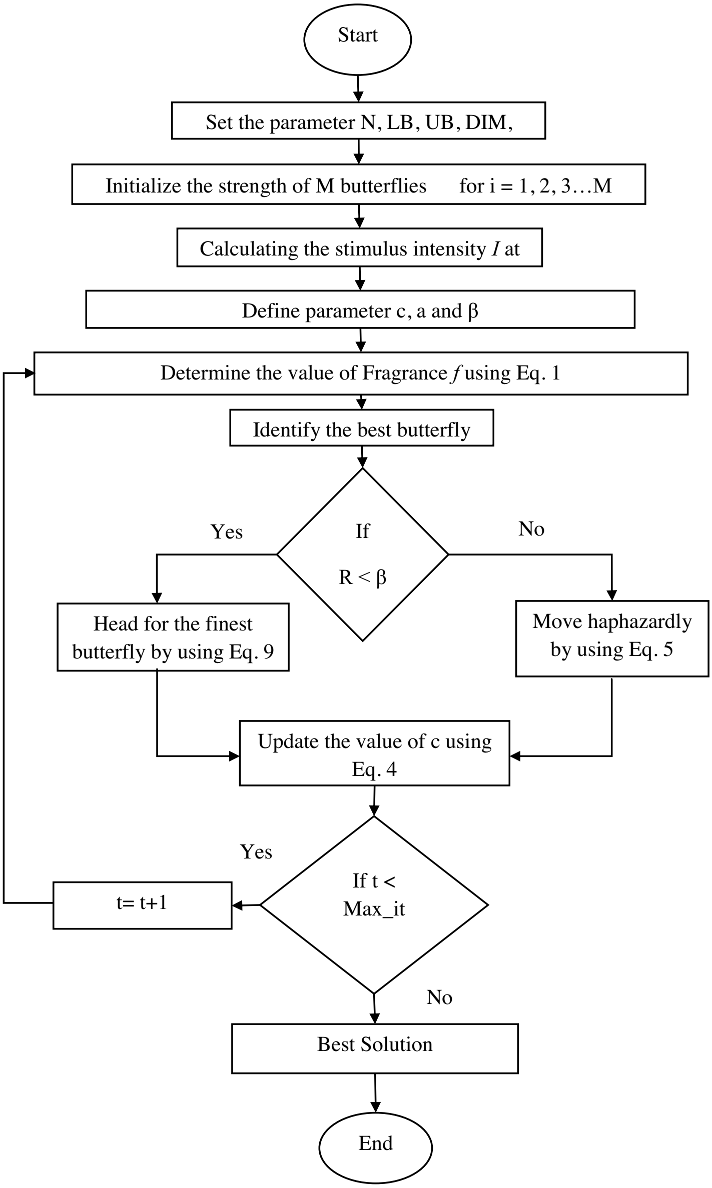
Start
Set the parameter N, LB, UB, DIM,
Initialize the strength of M butterflies for i = 1, 2, 3…M
Calculating the stimulus intensity *I* at
Define parameter c, a and β
Determine the value of Fragrance *f* using Eq. 1
Identify the best butterfly
If R < β
Yes
No
Head for the finest butterfly by using Eq. 9
Move haphazardly by using Eq. 5
Update the value of c using Eq. 4
If t < Max_it
Yes
No
t= t+1
Best Solution
End

Figure 1. *Illustrate the flowchart of IBOA technique.*

The barrier of the path is designed by using Eq. (12). If any point in the solution space comes into contact with the barrier, a penalty is added to the fitness function, and it is represented in Eq. (13), where p is the penalty function [38].

$$O_b = \frac{\sqrt{(X_i - X_o)^2 + (Y_i - Y_o)^2}}{R_o} - 1 \begin{cases} O_b > 0, \textit{without barrier} \\ O_b < 0, \textit{with barrier} \end{cases} \tag{12}$$

where (X_o, Y_o) *and* R_o represents the position of the center and radius of obstacle o, respectively, and $0 = 1, 2, 3, \ldots\ldots$

$$P = P + mean(min\,(O_b, 0)\,) \tag{13}$$

Table III. *Analysis of path traveled and time taken for navigation, based on simulation and experimental analysis of a single NAO.*

	Simulation Results		Experimental Results		% of Deviation	
Sl. No.	Path Travelled (cm)	Time Taken (sec)	Path Travelled (cm)	Time Taken (sec)	Path Travelled	Time Taken
1	287.23	39.83	309.24	43.01	7.12	7.39
2	288.14	39.96	309.31	43.16	6.84	7.41
3	287.67	39.89	310.76	43.09	7.43	7.43
4	287.52	39.87	310.51	43.05	7.40	7.39
5	288.86	40.06	309.83	42.95	6.77	6.73
6	288.79	40.05	309.72	42.83	6.76	6.49
7	288.26	39.98	310.45	43.18	7.15	7.41
8	287.59	39.88	310.58	43.86	7.40	9.07
9	287.85	39.92	309.91	43.22	7.12	7.64
10	288.53	40.01	310.43	43.78	7.05	8.61
Avg.	288.04	39.95	310.07	43.21	7.10	7.56

The objective function by using the penalty function for a barrier free path represents the shortest path length is mentioned in Eq. (14).

$$PL = D \times (1 + w \times P) \tag{14}$$

Where w represents the penalty factor with a value of 10. [38]

3. Application of IBOA in path navigation of NAO

3.1. Experimental results and discussion

The simulation investigation for path identification is performed using Webots platform. To do the simulation task, the developed algorithm is implemented in Webots and programed in MATLAB. A simulation platform measuring 230 x 150 units with variously shaped barriers is created in Webots to evaluate the efficacy of the suggested methods. Simulated navigation between two points, i.e., start and goal point, utilizing the improved butterfly optimization technique, is carried out for single humanoid robot in intricate terrain full of obstacles. The path travelled and time taken to complete the path for a single NAO during simulation and experiments are depicted in Table III. The simulation results of both single and multiple NAO are displayed in Figures 2 and 4, respectively. To compare simulation findings with the IBOA's efficiency, experimental platforms that mimic a simulation environment in a lab setting are created. The navigation testing results for one humanoid and two humanoids are portrayed in Figures 3 and 5, respectively.

The results comparison indicates that the suggested controller performed satisfactorily with few variations in the difficult environmental conditions. The results comparison indicates that the suggested controller performed satisfactorily with few variations in the difficult environmental conditions. The primary reason for the slight variation in the results is the conditions considered during the movements of the manipulator from a particular position to a desired location. The ideal condition is considered during travel of the robot in the simulation platform, but because of the several factors such as floor smoothness, irregular route, friction between leg and floor, and internet connectivity, it is not taken into account while movement in the experimental platform.

Table IV shows the path length travelled for multiple humanoids to complete navigation during simulation and experiment.

Table V shows the time required for multiple humanoids to complete navigation during simulation and experiment.

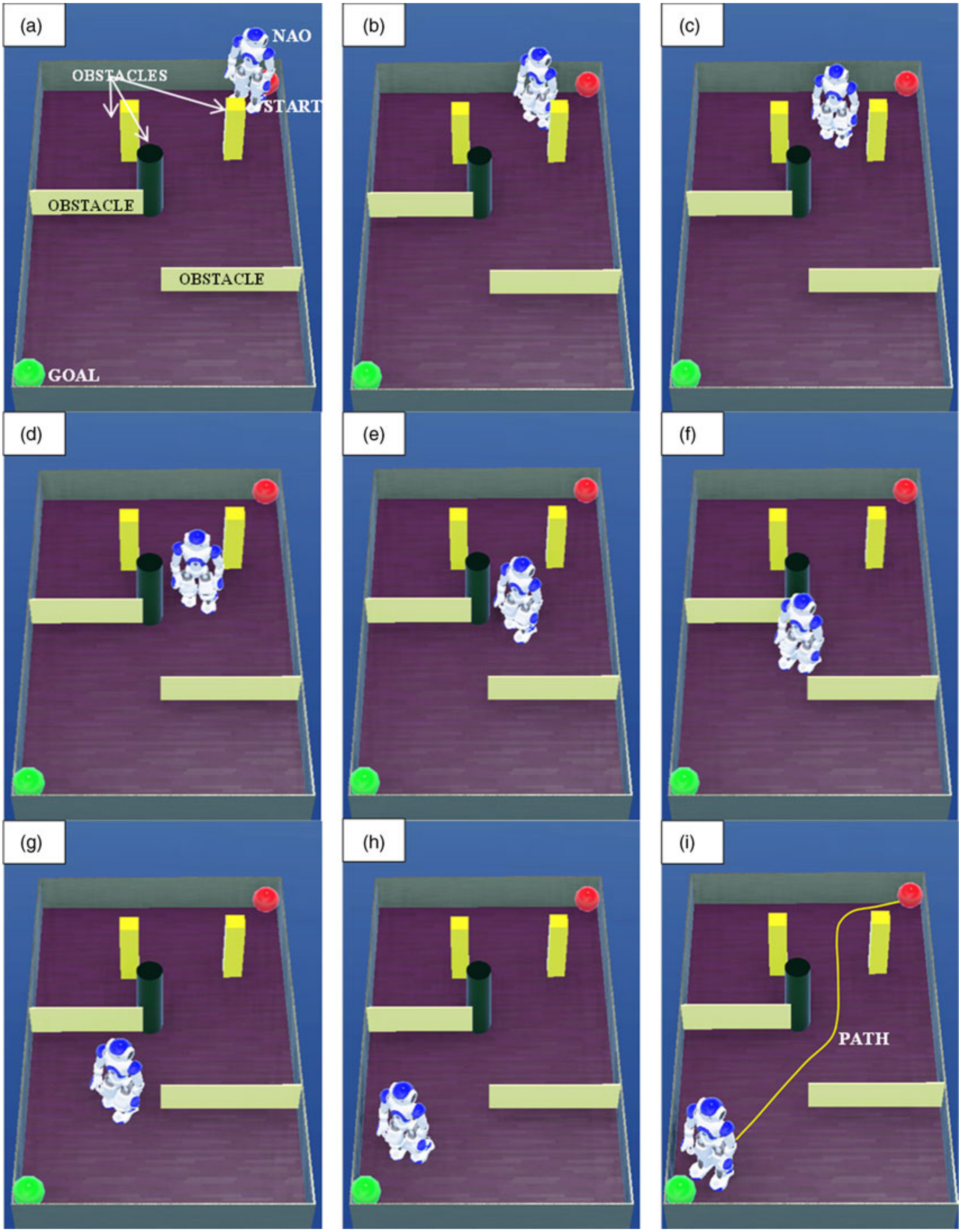

Figure 2. *Illustrate the Webots simulation result of single humanoid robot.*

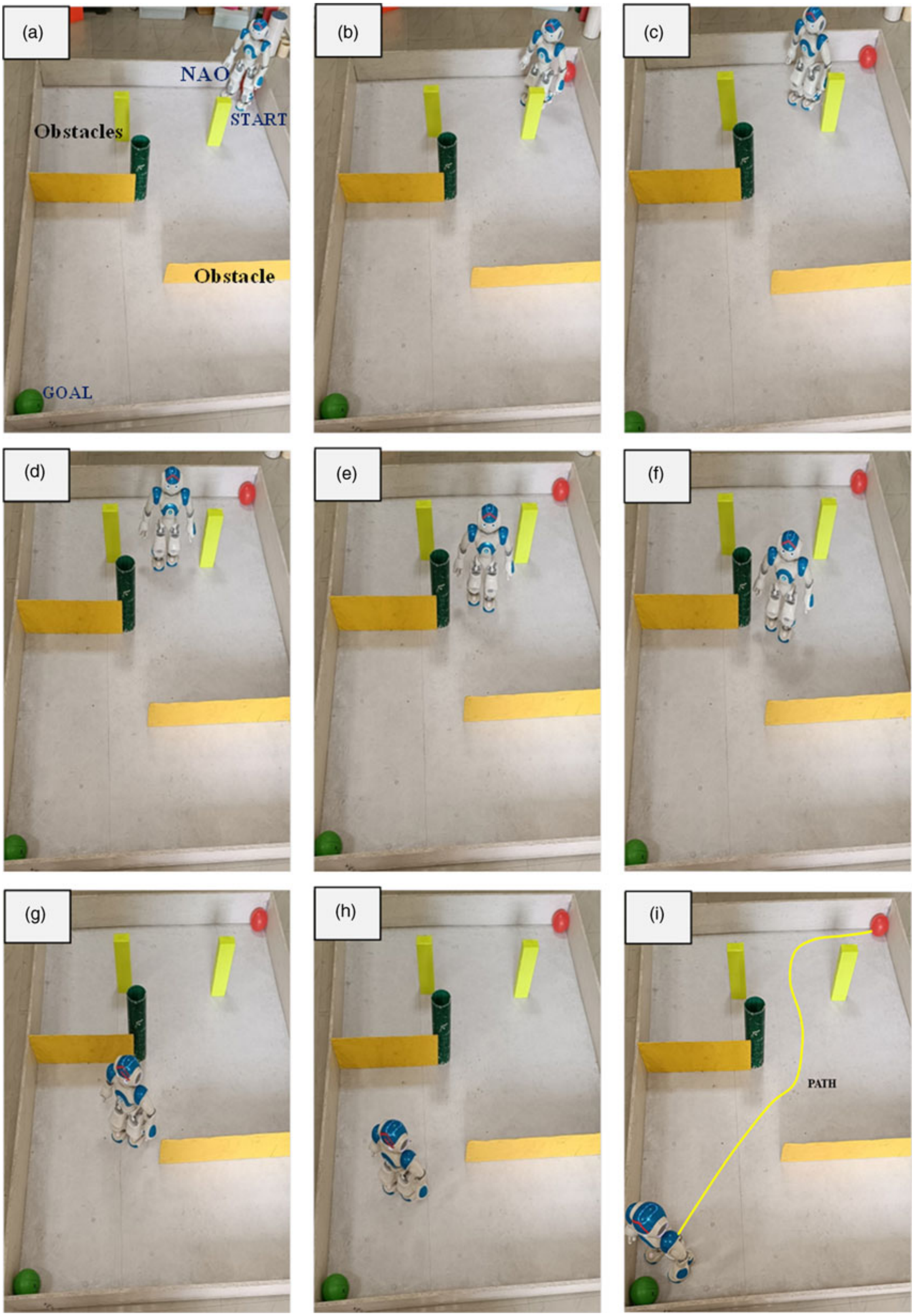

Figure 3. *Portrayed the experimental results of single humanoid NAO.*

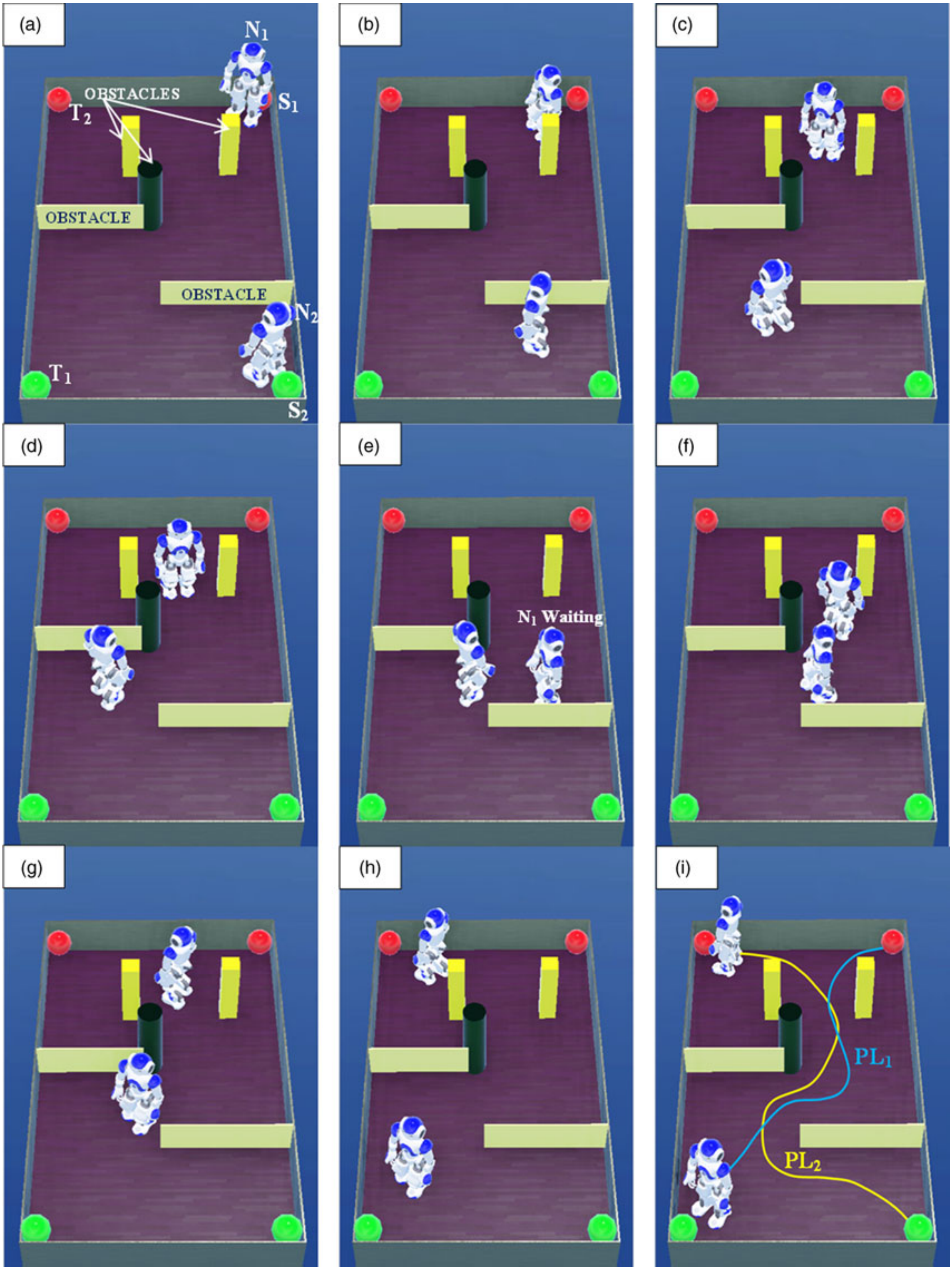

Figure 4. *Illustrate the Webots simulations results of multiple NAO.*

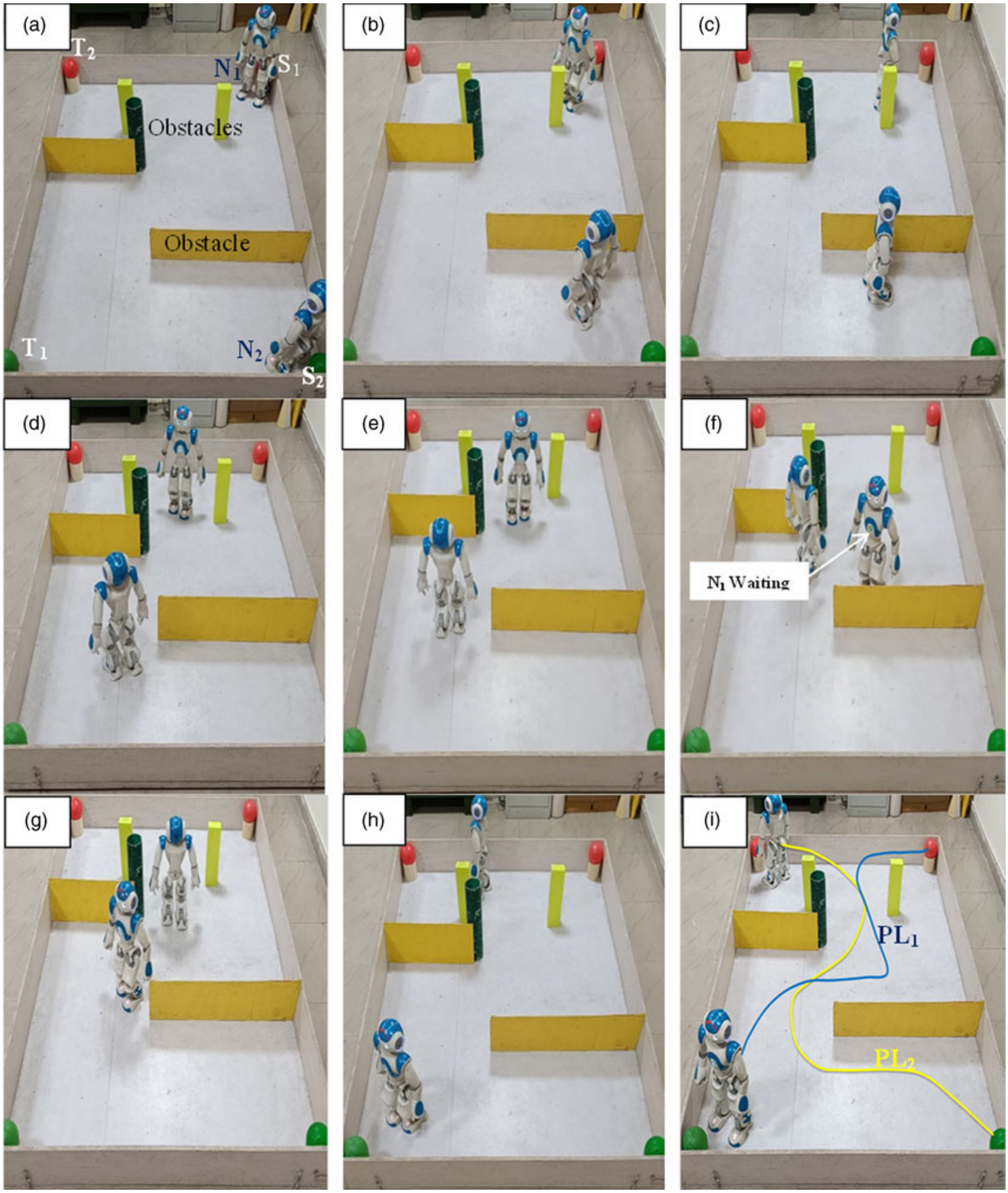

Figure 5. *Illustrate the real laboratory experimental outcomes of multiple NAO.*

4. Statistical evaluation based on the simulation and experimental studies

This section represents the histogram, probability, normality, and surface plots among the virtual simulation and real laboratory data acquired from the Webots simulation and real experimentation of humanoid NAO. Through statistical analysis, the investigation of different simulation and experimental data is performed. Through probability plot, normality plot, and histogram plot, the deviations of simulation results and experimental results can be easily observed. Surface plot is the 3D representation of relationship between navigation parameters.

Table IV. *Comparison of simulated and experimentally measured path lengths in the navigation of multiple NAO.*

	Simulation Path Travelled (cm)		Experimental Path Travelled (cm)		Deviation in Path Traveled (%)	
Sl. No.	**N1**	**N2**	**N1**	**N2**	**N1**	**N2**
1	302.34	310.75	318.25	327.64	5.00	5.16
2	302.13	309.21	317.37	326.52	4.80	5.30
3	301.56	310.38	317.78	326.85	5.10	5.04
4	301.89	310.84	317.86	326.79	5.02	4.88
5	301.75	310.43	318.59	327.18	5.29	5.12
6	302.45	309.52	317.52	327.89	4.75	5.60
7	302.25	310.61	317.39	327.47	4.77	5.15
8	301.76	309.69	318.53	326.69	5.26	5.20
9	301.54	309.78	317.48	326.68	5.02	5.17
10	302.26	309.91	318.83	326.83	5.20	5.18
Average	301.99	310.11	317.96	327.05	5.02	5.18

Table V. *Comparison of simulated and experimentally measured time required in the navigation of multiple humanoids.*

	Simulation Time Taken (sec)		Experimental Time Taken (sec)		Deviation in in Time taken (%)	
Sl. No.	**N1**	**N2**	**N1**	**N2**	**N1**	**N2**
1	41.25	39.78	43.58	41.82	5.35	4.88
2	42.54	38.43	44.82	40.35	5.09	4.76
3	41.78	38.25	43.76	40.17	4.52	4.78
4	41.92	39.89	43.11	40.28	2.76	0.97
5	42.26	39.76	44.85	41.95	5.77	5.22
6	42.35	38.54	43.91	40.82	3.55	5.59
7	42.11	38.28	44.27	41.38	4.88	7.49
8	41.79	38.39	43.57	40.76	4.09	5.81
9	41.68	39.76	44.42	40.75	6.17	2.43
10	41.76	38.19	43.83	40.52	4.72	5.75
Average	41.94	38.93	44.01	40.88	4.69	4.77

4.1. Statistical evaluation of results of single NAO

The trial data for both simulation and real-world experiments of single humanoid NAO are presented in this section.

4.1.1. Histogram plots

The histogram plots of virtual and real laboratory results obtained by using single humanoid robot are presented in Figure 6. The plots are shown in two categories: the simulation and experimental path travelled, represented in Figure 6(a), while the simulation required time and experimental required time by a single humanoid are represented in Figure 6(b). The mean value and standard deviation are shown in the histogram plots for both the path length and required time by single humanoid to reach the goal point.

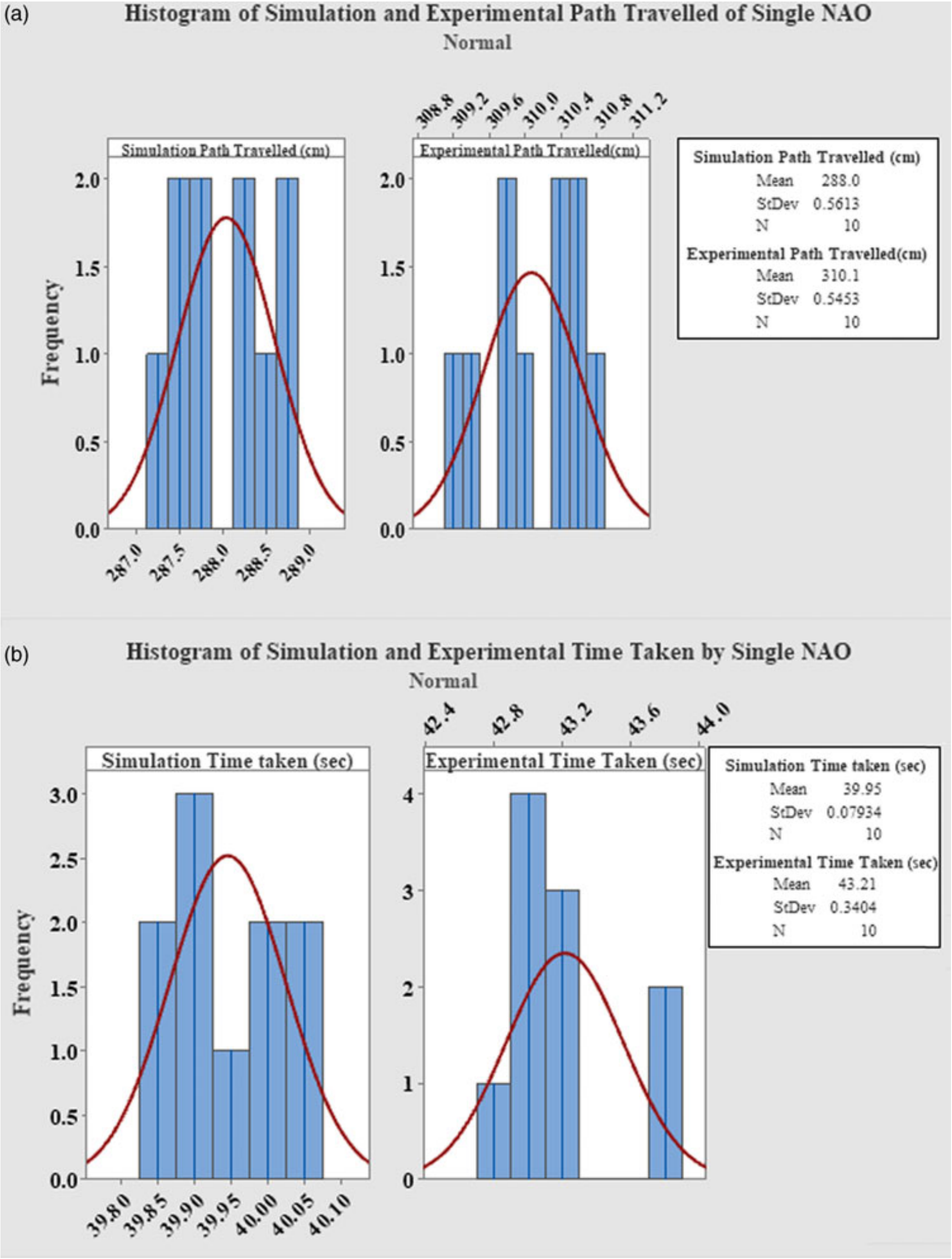

Figure 6. *Histogram plot of virtual and experimental (a) path travelled (b) Time taken by single NAO.*

4.1.2. Normality plots

The Anderson-Darling method has been followed to perform the normality test. The test outcomes are presented in Figure 7. The tests are performed separately for each parameter, i.e, simulation and experimental path traveled and time taken. The mean, standard deviation, P-value, etc. are shown in the graph.

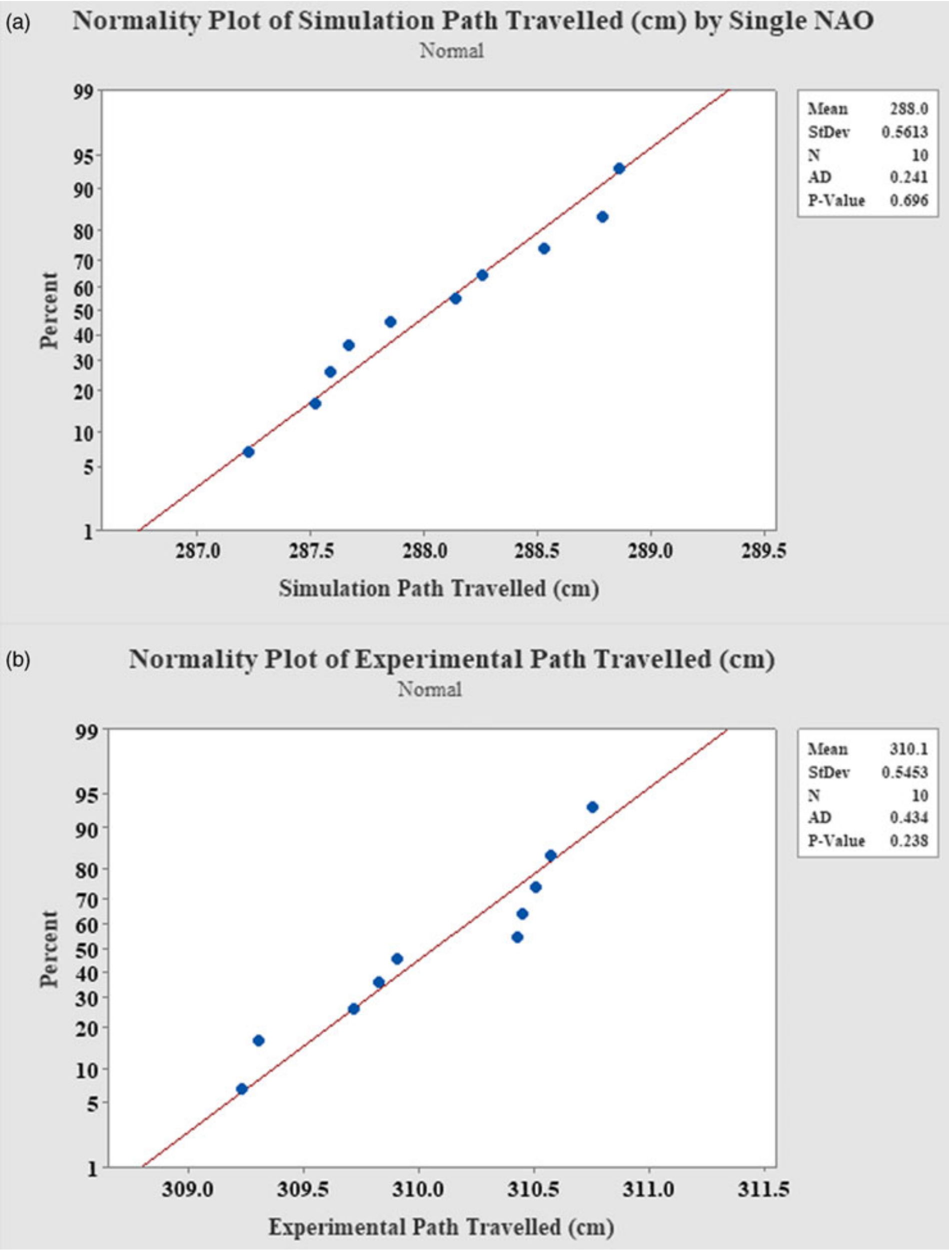

Figure 7. *Normality plot of (a) Simulation path travelled (b) Experimental path travelled (c) Simulation time taken (d) Experimental time taken by single NAO.*

4.1.3. Probability plots

The probability plots of virtual and real laboratory path travelled are shown in Figure 8(a), whereas the probability plot of virtual and experimental time consumed by the single NAO is shown in Figure 8(b). The curve follows the normal distribution with 95% confidence intervals. The p-Value, mean, andstandard deviation of all the parameters are shown in the plots.

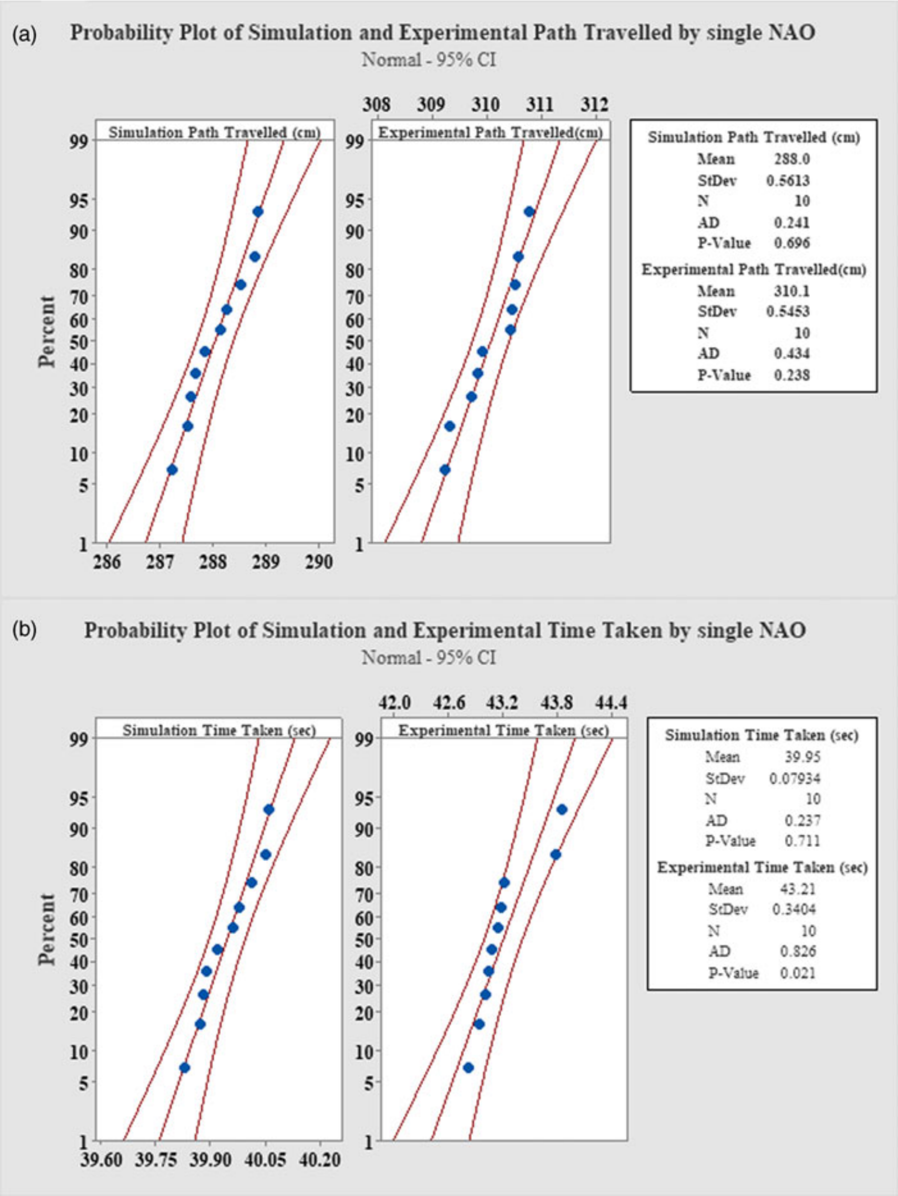

Figure 8. *Probability plots of simulation and experimental (a) path travelled (b) time taken by using single NAO.*

4.1.4. Surface plots

The surface plots among simulation and experimental path travelled and deviation in path travelled are presented in Figure 9(a). The time consumed in virtual and experimentation and the percentage variation in time taken are shown in Figure 9(b).

4.2. Statistical evaluation of results of multiple NAO

The experimental data for both simulation and real-world experimentation of multiple NAOs are presented in this section. The notations used in the analysis are mentioned below.

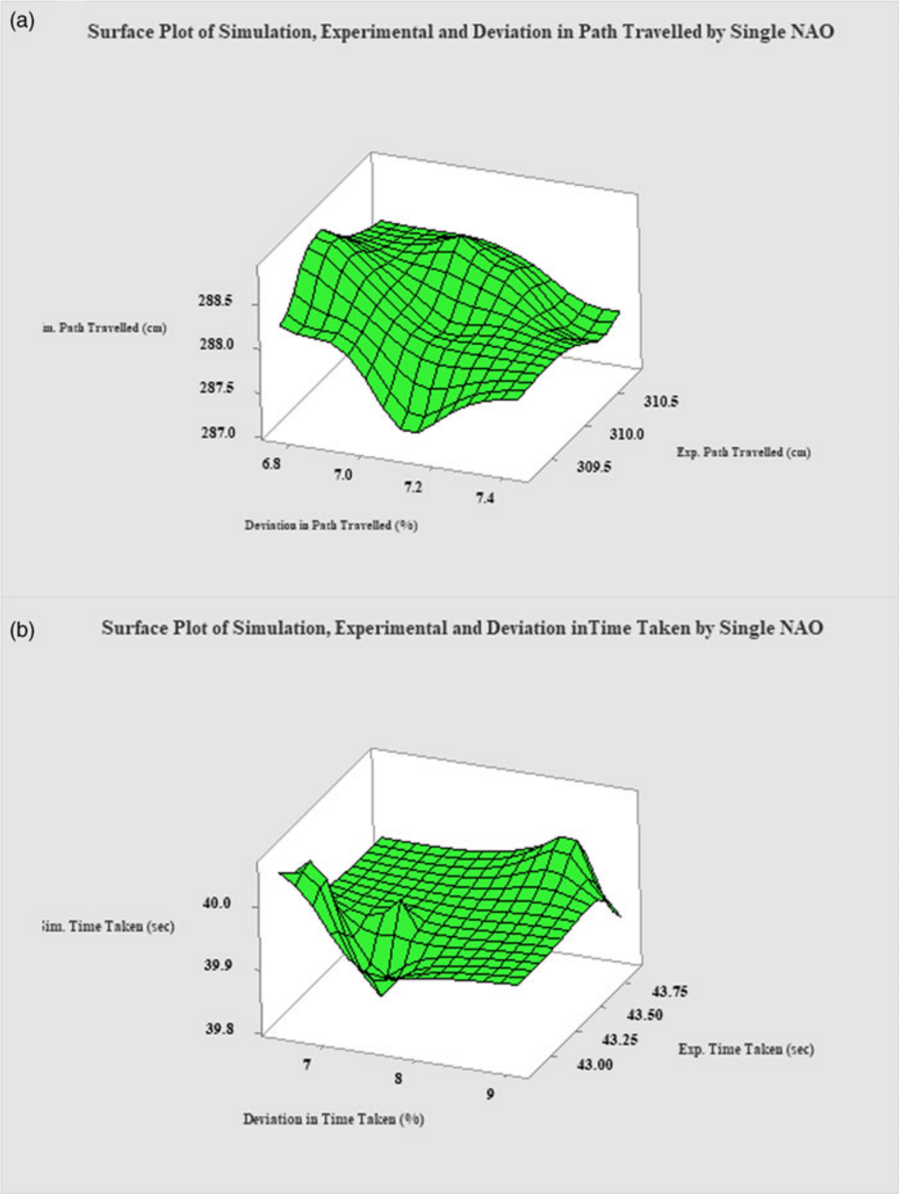

Figure 9. *(a) Surface plot of simulation, experimental and deviation in path travelled obtained by using single NAO (b) Simulation, experimental and deviation in time taken by using single NAO.*

N1: Humanoid NAO 1
N2: Humanoid NAO 2
PT: Path Travelled
TT: Time Taken
SPT: Simulation Path Travelled
EPT: Experimental Path Travelled
STT: Simulation Time Taken
ETT: Experimental Time Taken
DPT: Deviation in Path Travelled
DTT: Deviation in Time Taken

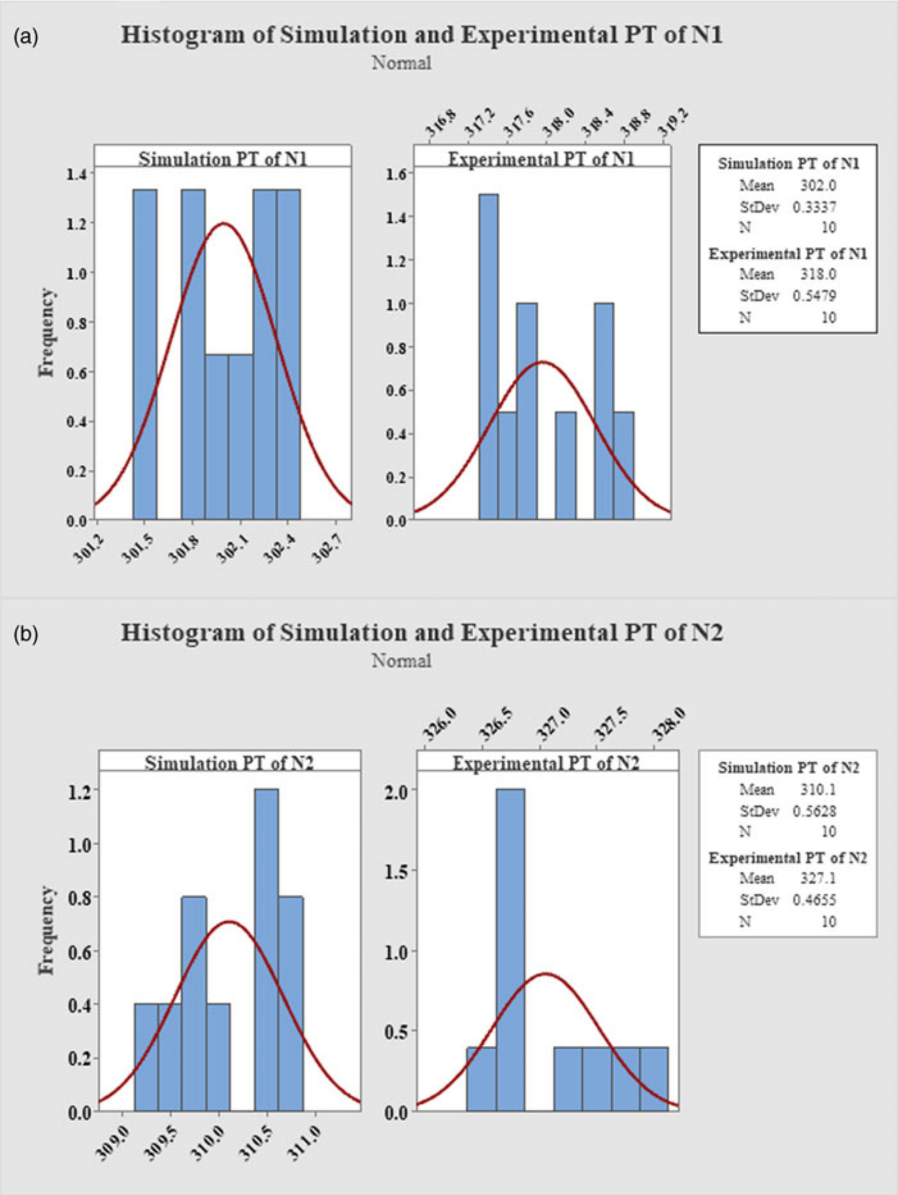

Figure 10. *Histogram plots of (a, b) SPT and EPT obtained by N1 and N2 (c, d) STT and ETT obtained by N1 and N2.*

4.2.1. Histogram plots

The histogram plots of virtual and real experimentation result obtained by using multiple humanoid NAO robots are presented in Figure 10. The plots have shown in two categories, the simulation and experimental path traveled of N_1 and N_2 exhibited in Figure 10(a) and (b) while the simulation required time and experimental required time by N_1 and N_2 represented in Figure 10(c) and (d). The mean value and standard deviation are shown in the histogram plots for both the path traveled and time taken by N_1 and N_2 to reach the goal point.

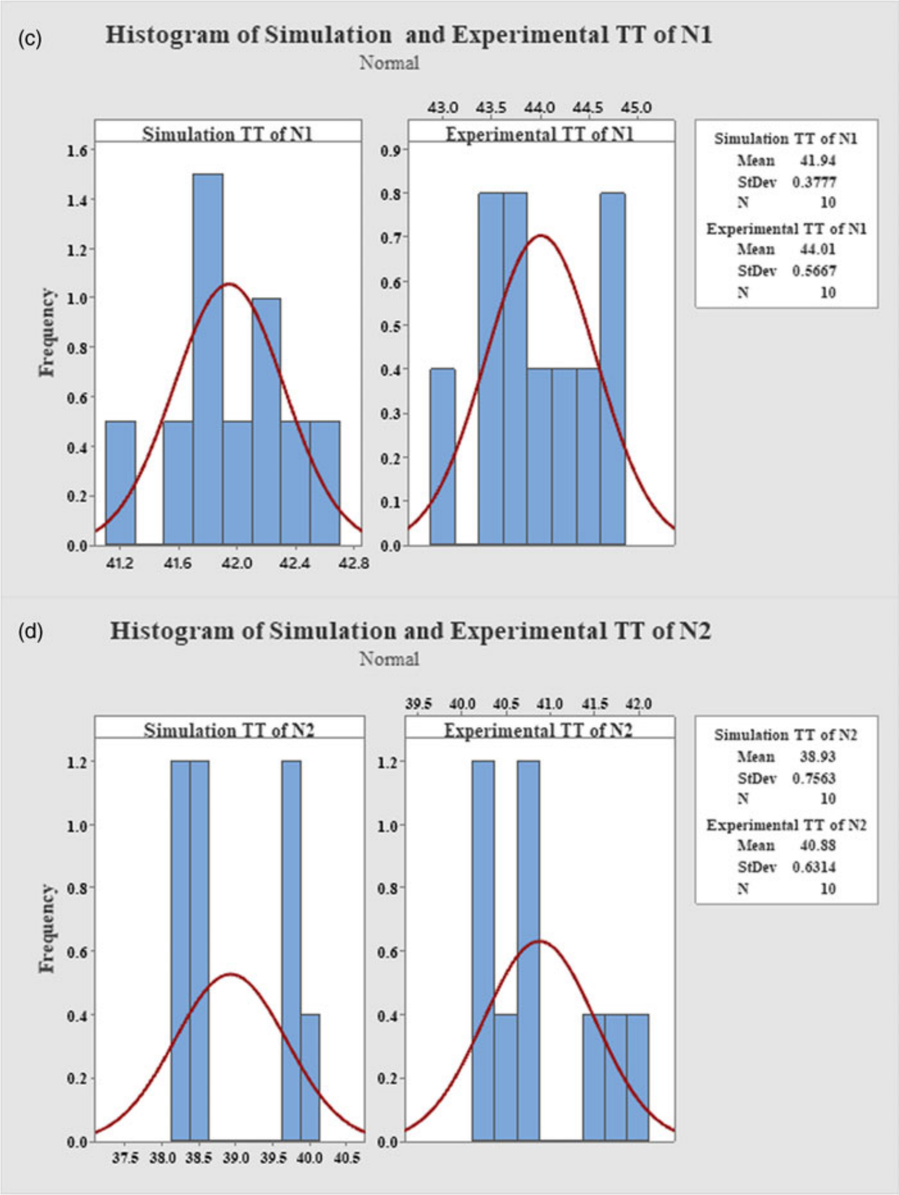

Figure 10. *continued.*

4.2.2. Normality plots

The Anderson-Darling method has been followed to perform the normality test. The test results are shown in Figures 11 and 12. The tests are performed separately for each parameter, i.e, SPT, EPT, STT & ETT by N_1 and N_2. The mean, standard deviation, P-value, etc. are shown in the graph.

4.2.3. Probability plots

The probability plots of SPT and EPT outcomes are displayed in Figure 13(a) and (b), whereas the probability plots of STT and ETT results obtained by the N_1 and N_2 robots are potrayed in Figure 13(c) and (d). The curve follows the normal distribution with 95% confidence intervals. The p-Value, mean, and standard deviation of all the parameter are shown in the plots.

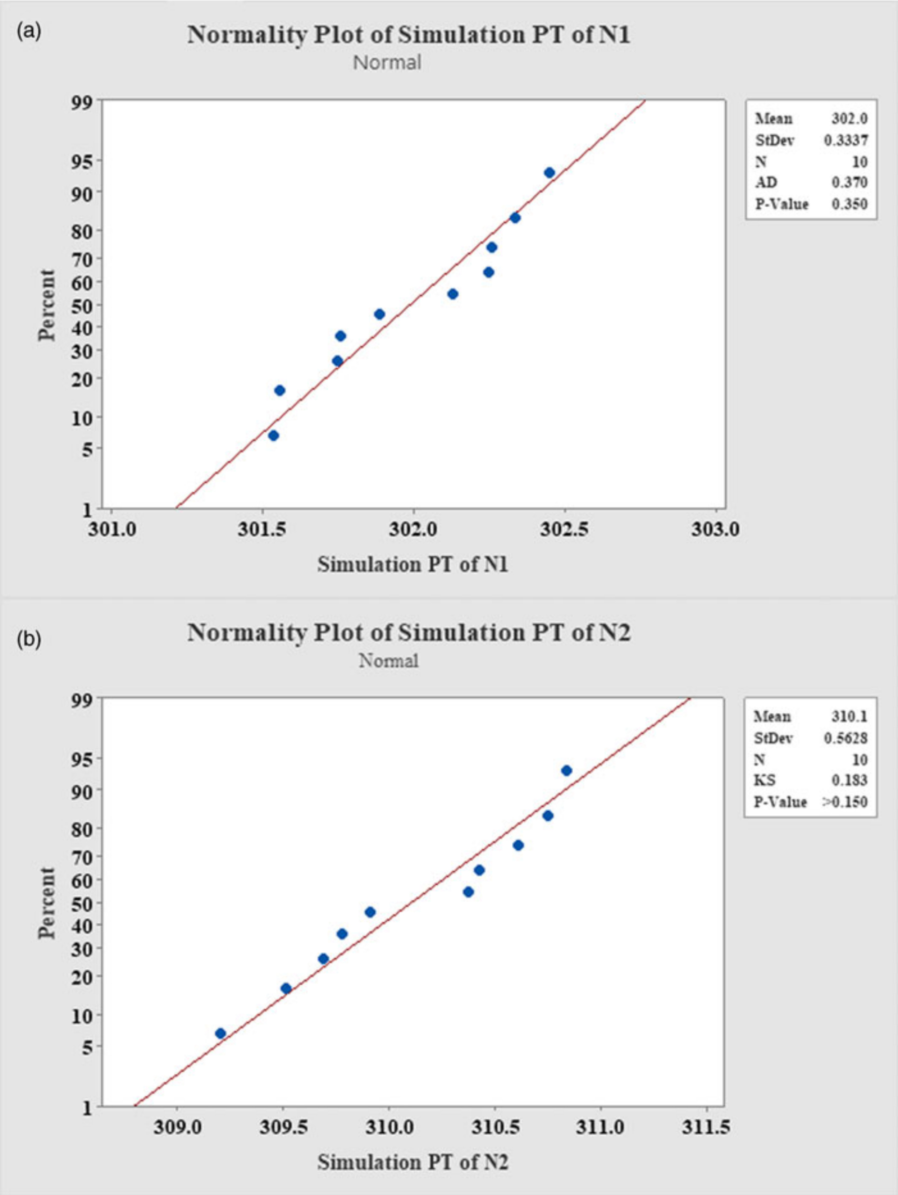

Figure 11. *(a, b) Normality plots of SPT results of N1 and N2. (c, d) Normality plots of EPT results of N1 and N2.*

4.2.4. Surface plots

The surface plots among SPT, EPT, DPT and STT, ETT, DTT results obtained by N1 and N2 are displayed in Figure 14.

5. Evaluation of the IBOA approach in comparison to other established path planning methods

The efficiency of the BOA model was assessed by comparing it to another motion planning model developed by Wang et. al. [38]. A two-dimensional robotic environment, as presented by the author, is designed in MATLAB with dimensions set to 6x6. The start and goal points are set to (0, 0) and (4, 6),

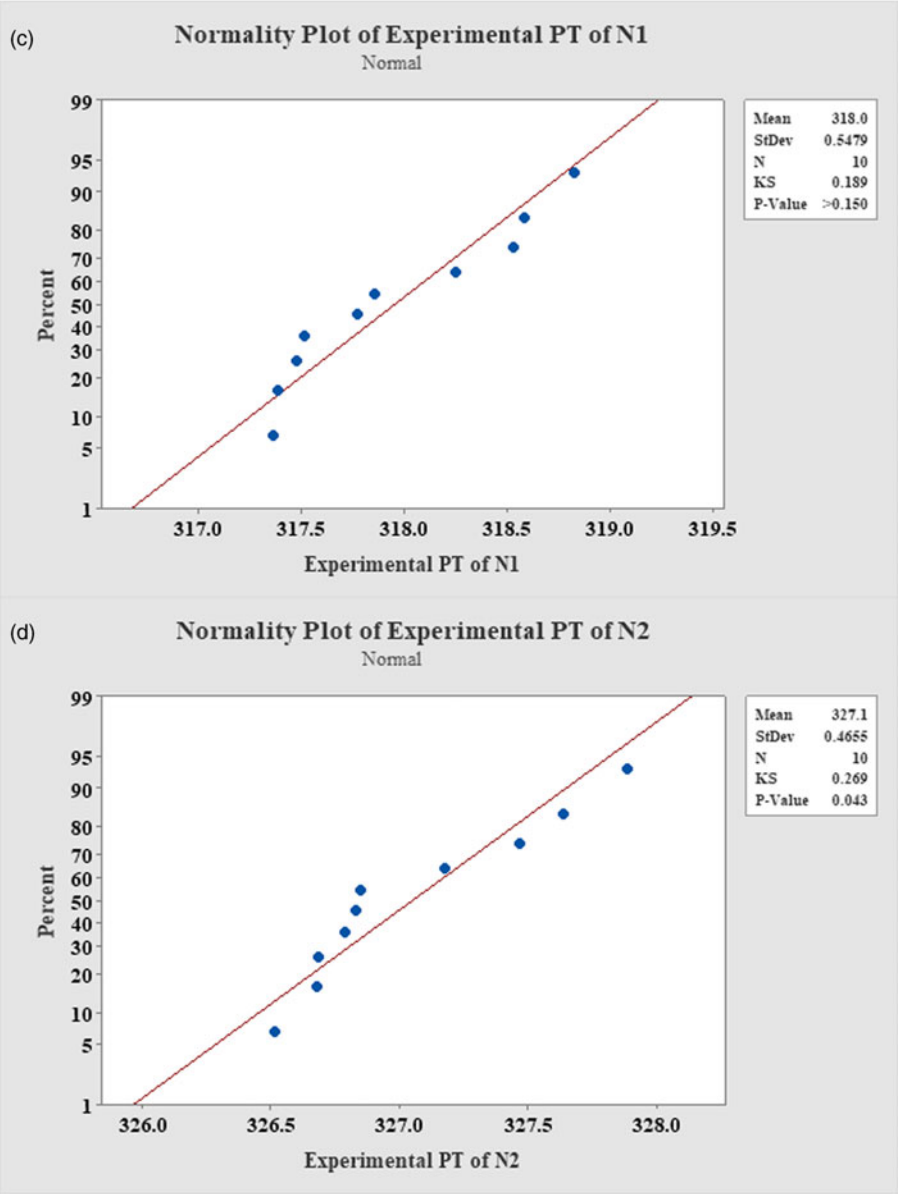

Figure 11. *continued.*

respectively. The environment is modeled with 7 obstacles for the simulation in MATLAB, which is shown in Figure 15. The comparison of the convergence curve is shown in Figure 16. Table VI illustrates the comparison results between the proposed models and other existing models such as Adaptive parallel arithmetic optimization algorithm (APAOA). The effectiveness of the outlined IBOA is again evaluated with another model, Fuzzy ACO (FACO), proposed by Yen and Cheng [39], and the results are shown in the Figures 17 and 18. The comparison results with FACO is presented in Table VII.

The results indicate that the proposed model achieves a substantial boost in efficiency such as 7.4% improvement compared to the existing navigational model APAOA used by the author Wang et. al. [38] and 5.47% compared to FACO model implemented by the author Yen and Cheng [39].

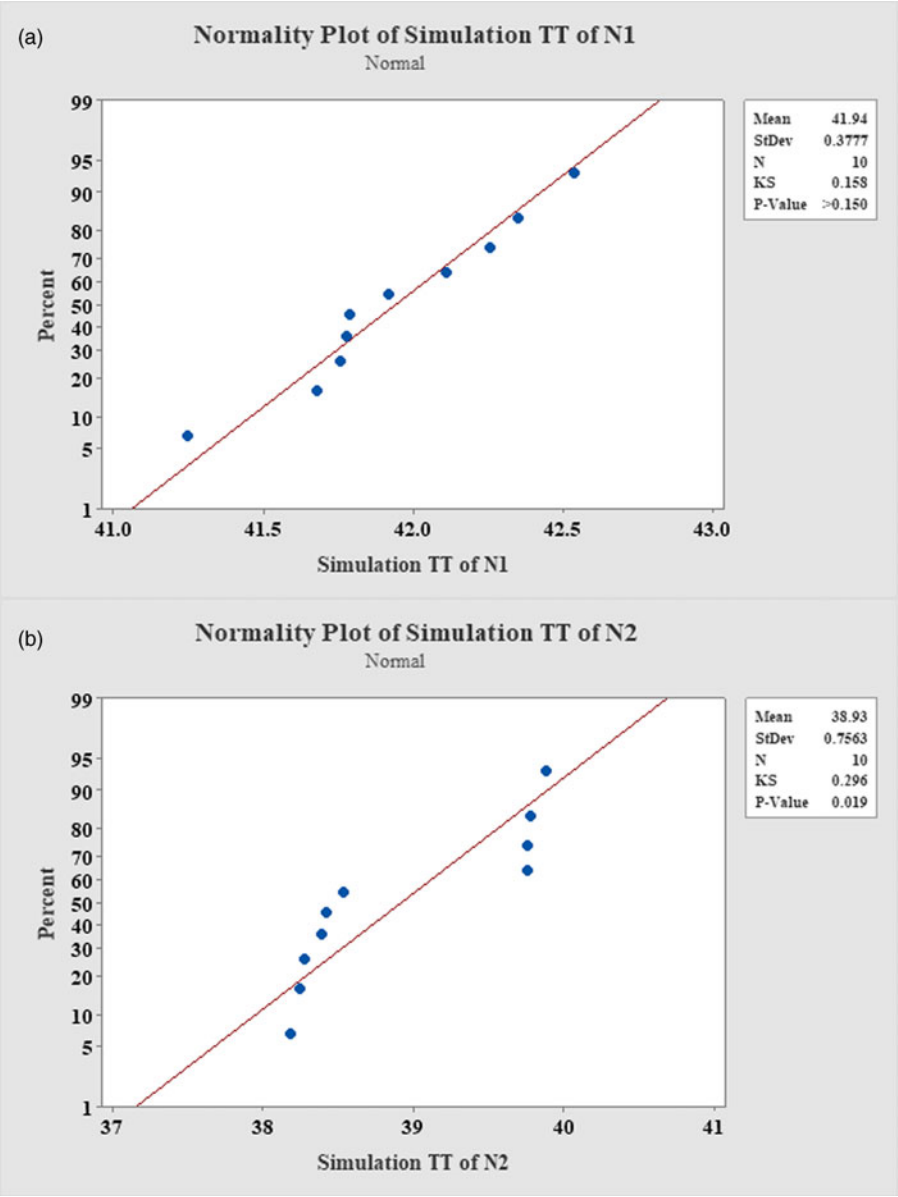

Figure 12. *(a, b) Normality plots of STT results of N1 and N2 (c, d) Normality plots of ETT results of N1 and N2.*

6. Conclusion

The suggested IBOA approach is designed in the current studies for the humanoid robot navigation and motion planning under challenging environmental condition. The suggested algorithm is tested on the single and multiple humanoids in Webots simulator, and the results are evaluated with the outcomes of experimental setup. It is found that the robot effectively follows the path in both environments. To demonstrate the efficacy of the suggested method for the motion planning of the legged manipulator

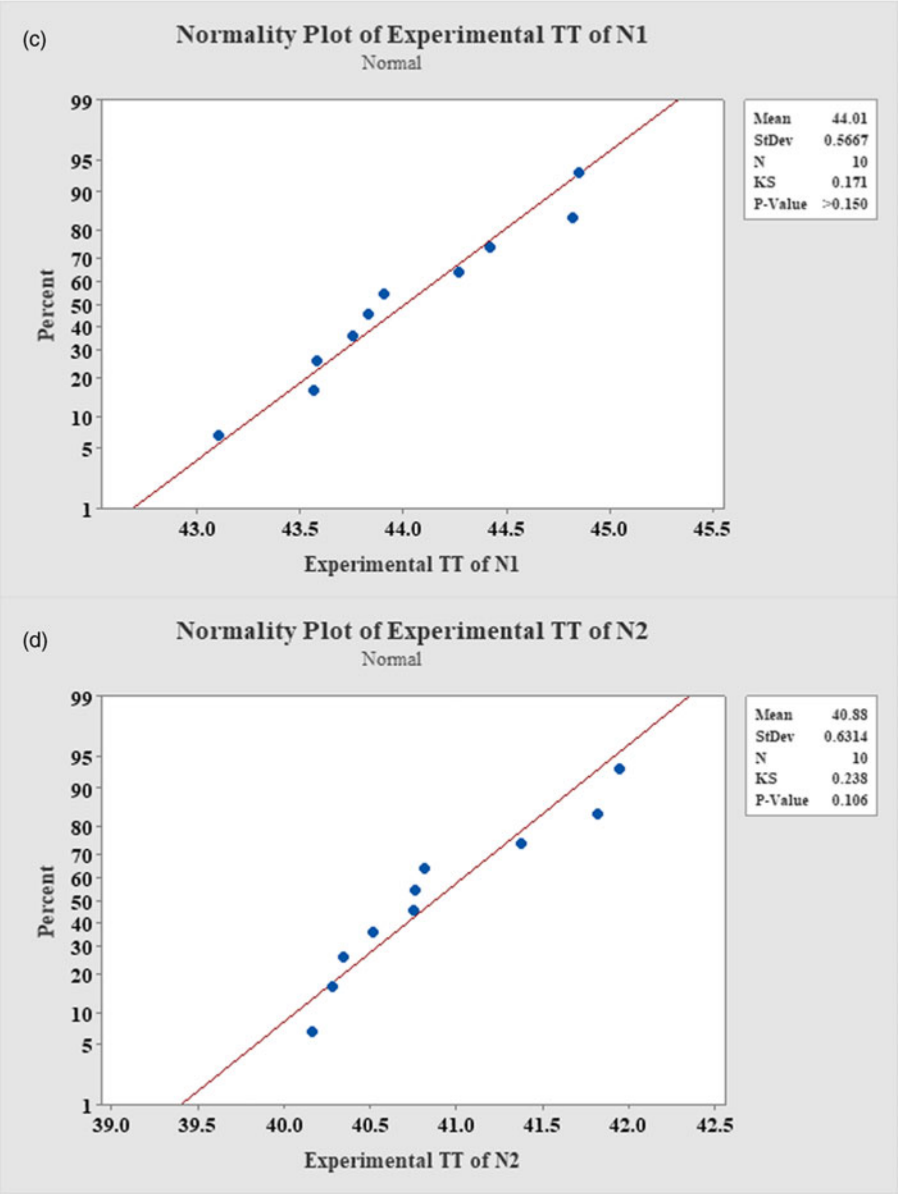

Figure 12. *continued.*

in unclear situations, statistical analyses such as histogram plots, probability plots, normality plots, and surface plots are conducted. The proposed IBOA technique has shown notable improvements in efficiency compared to conventional methods, demonstrating significant enhancements. Two different models, such as APAOA and FACO, have been referred to evaluate the potential of the proposed IBOA. An improvement of 7.4% is achieved compared to APAOA, and 5.47% is achieved when comparing with FACO model. The convergence curve of both environments using IBOA is presented, and it is found that the proposed IBOA model successfully finds the shortest path by avoiding obstacles during navigation.

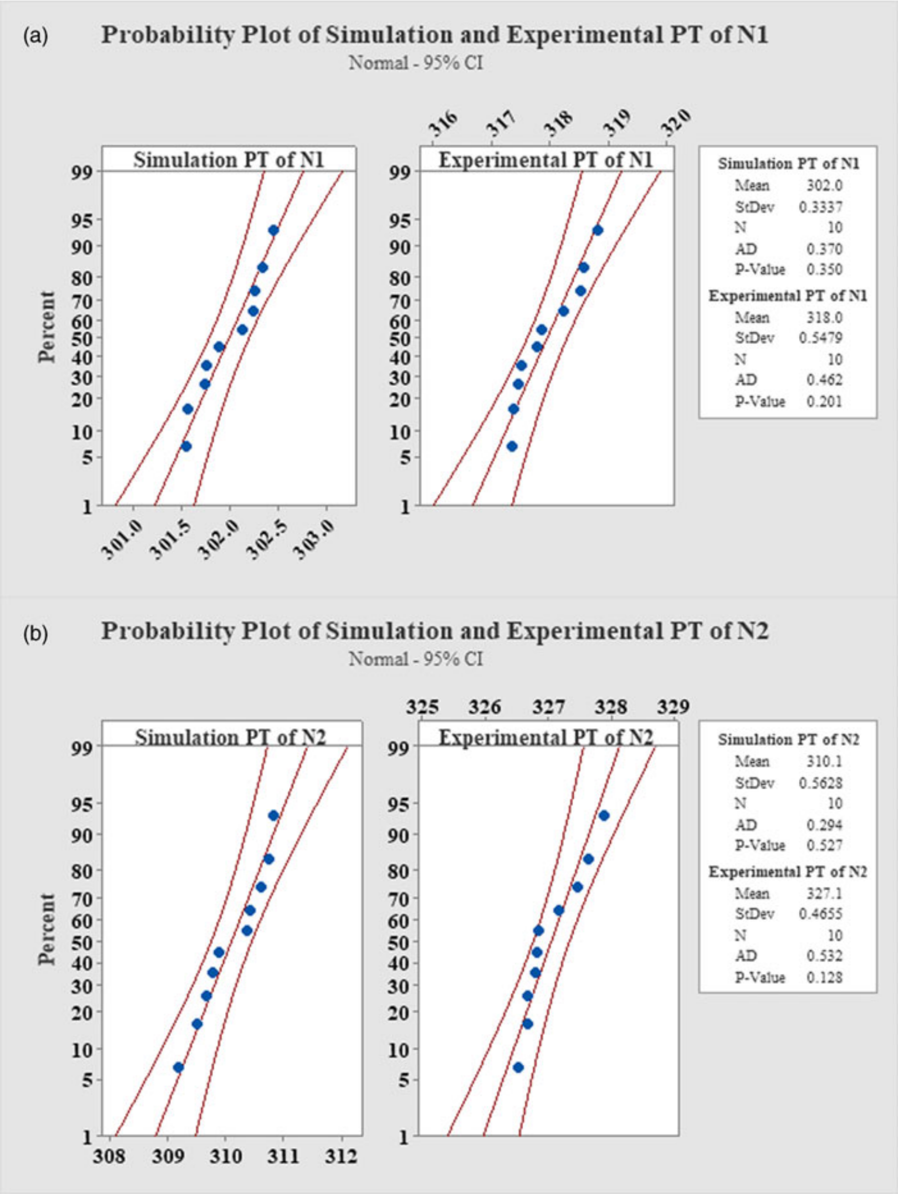

Figure 13. *(a, b) Probability plots of SPT and EPT results of N1 and N2 (c, d) Probability plots of STT and ETT results of N1 and N2.*

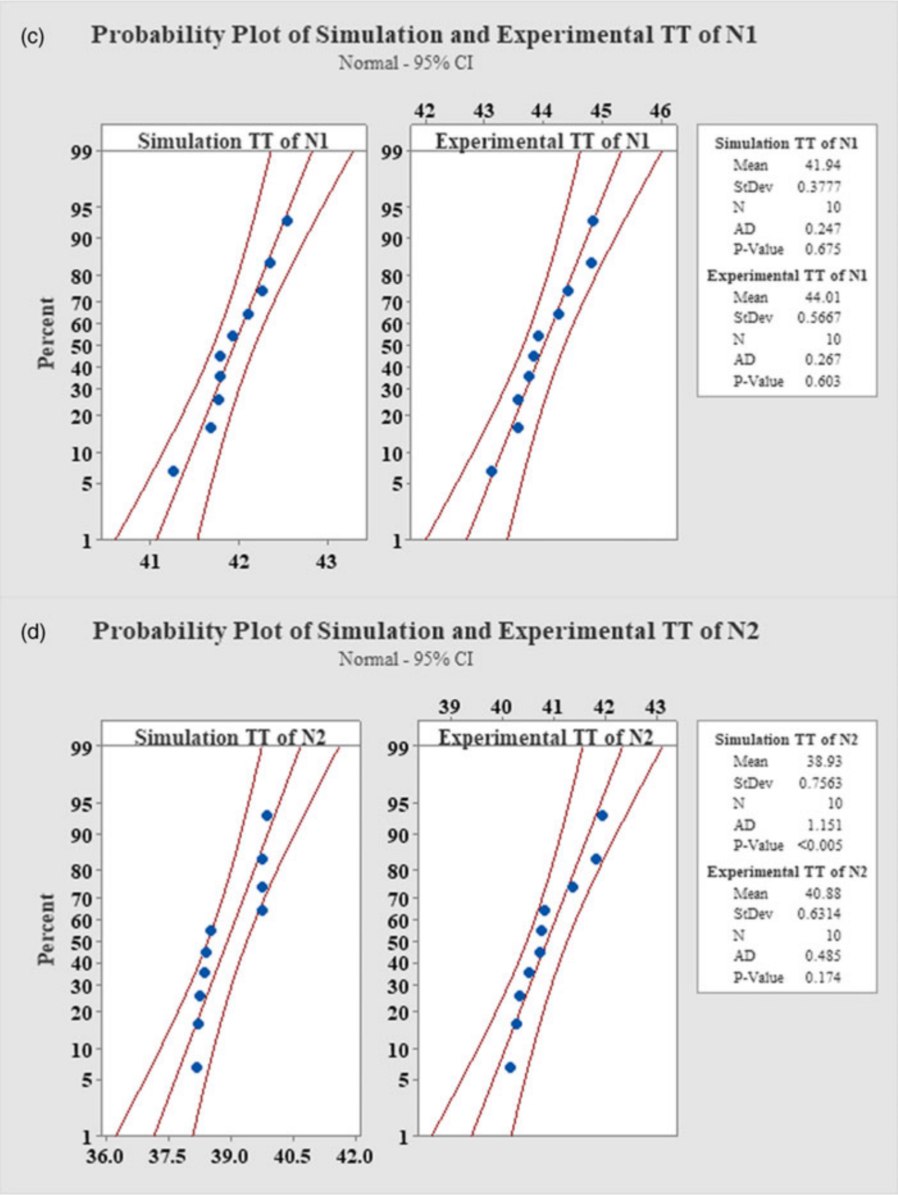

Figure 13. *continued.*

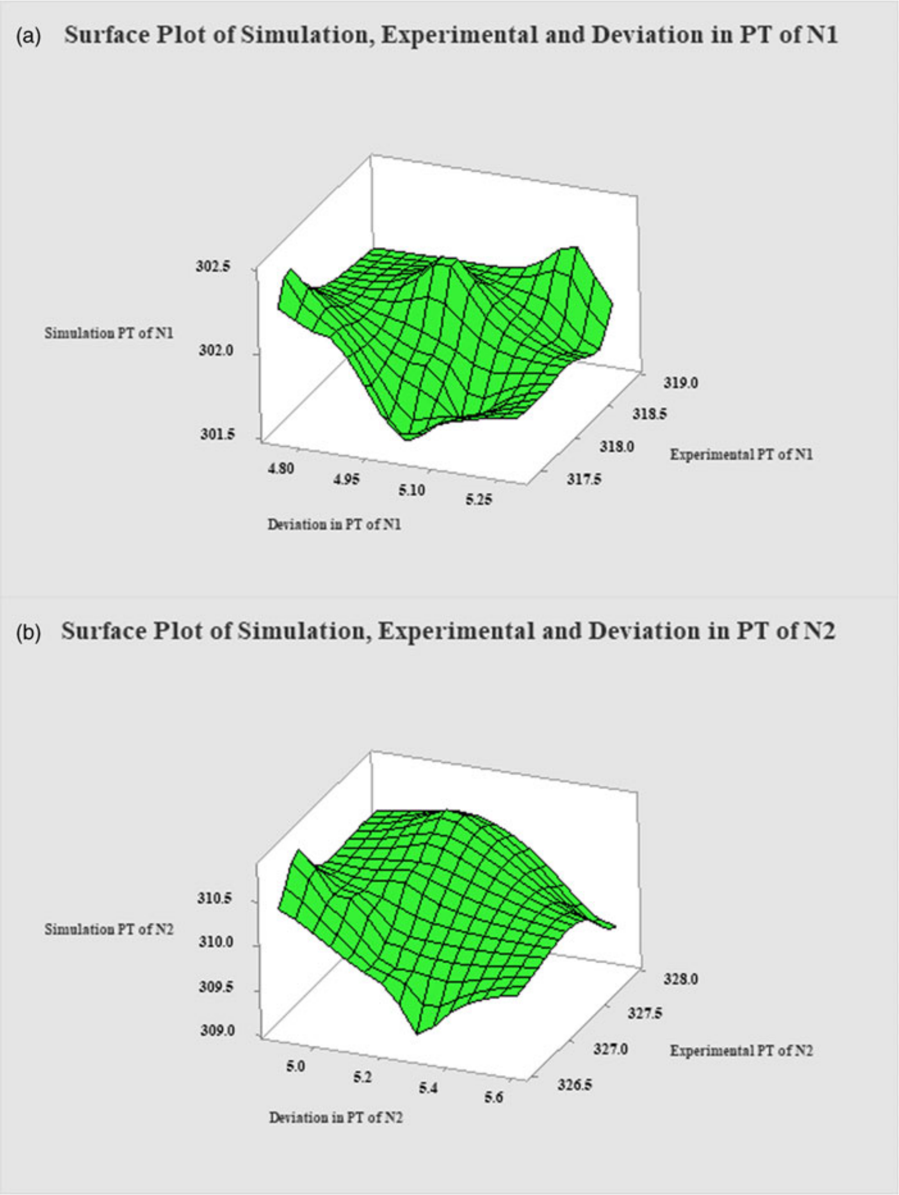

Figure 14. *(a, b) Surface plots among SPT, EPT, and DPT results of N1 and N2. (c, d) Surface plots among STT, ETT, and DTT results of N1 and N2.*

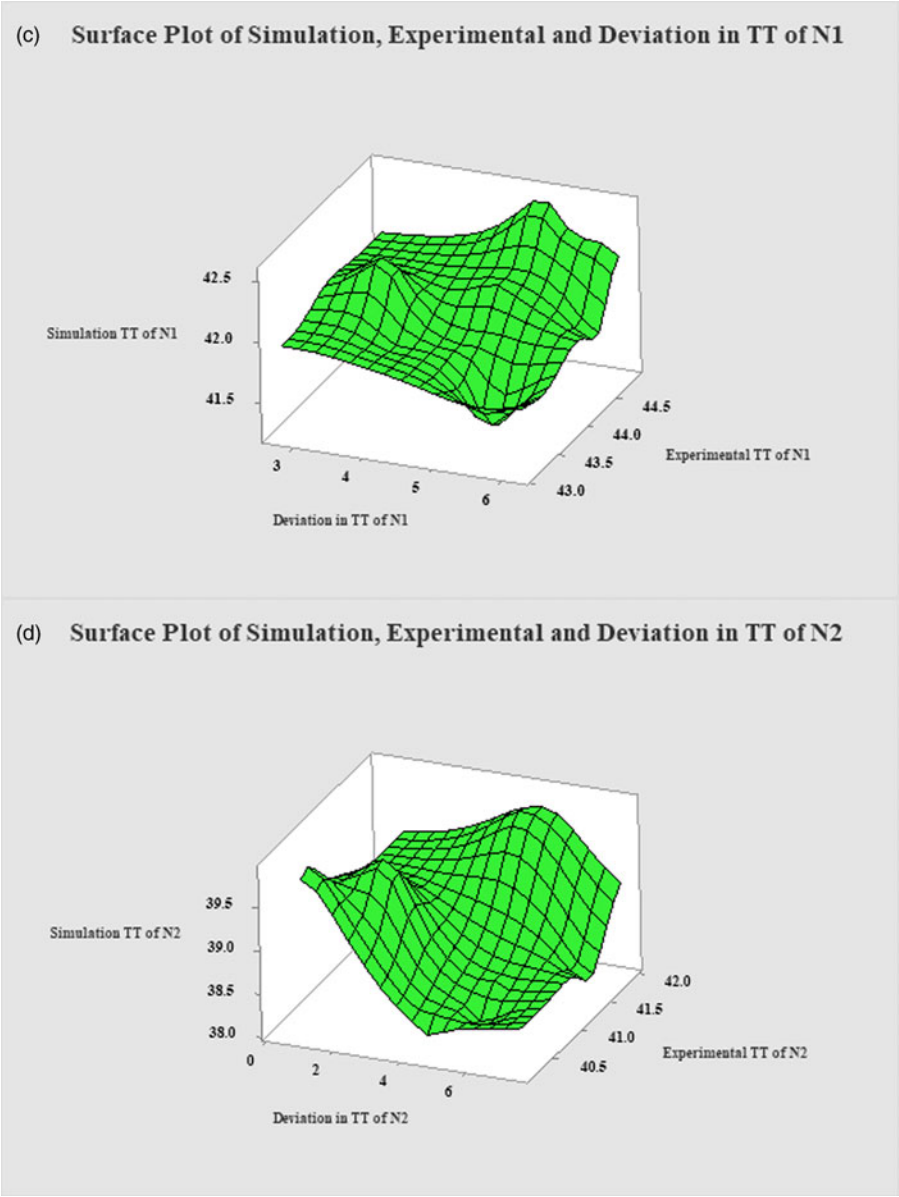

Figure 14. *continued.*

Table VI. *Comparison between improved butterfly optimization algorithm (IBOA) and APAOA used by Wang et. al. [27] in scene 2.*

Model Used	**Path Traveled (in units)**	**% Deviation**
APAOA (Figure 9) Wang et. al. [27]	7.40	**7.4**
IBOA	**6.89**	

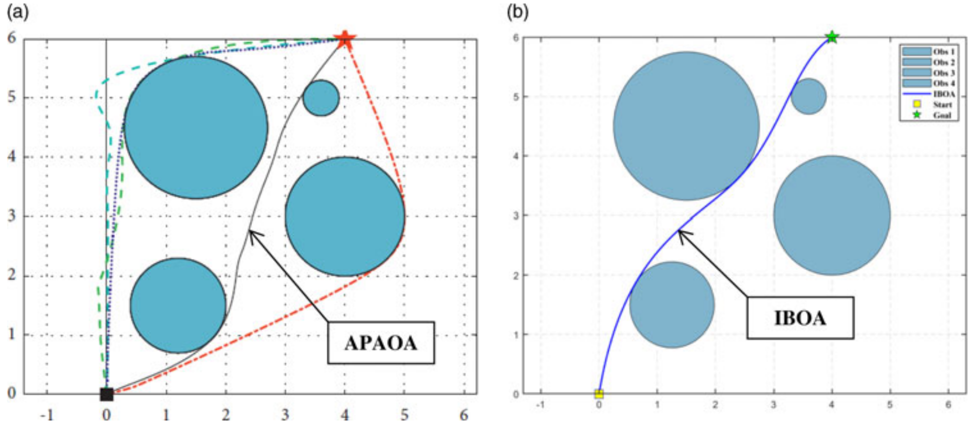

Figure 15. *Trajectory achieved by (a) Wang et. al. [27] and (b) improved butterfly optimization algorithm in identical environment.*

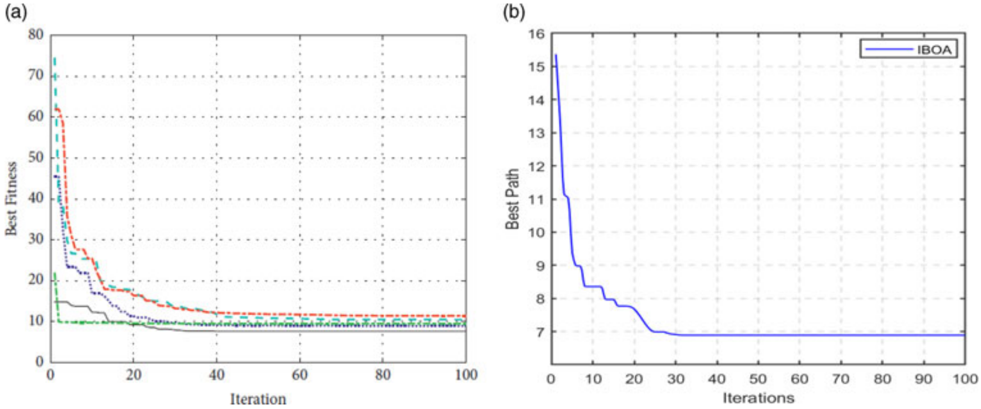

Figure 16. *(a) Convergence curve generated by Wang et. al. [27]. (b) Convergence curve generated by the proposed improved butterfly optimization algorithm model in scene 2.*

Table VII. *Comparison between the improved butterfly optimization algorithm (IBOA) and Fuzzy ACO (FACO) implemented by Yen and Cheng [39].*

Model Used	Path Traveled (in units)	% Deviation
FACO (Yen and Cheng [39])	29.3848	**5.47**
IBOA (Figure 18(b))	**27.86**	

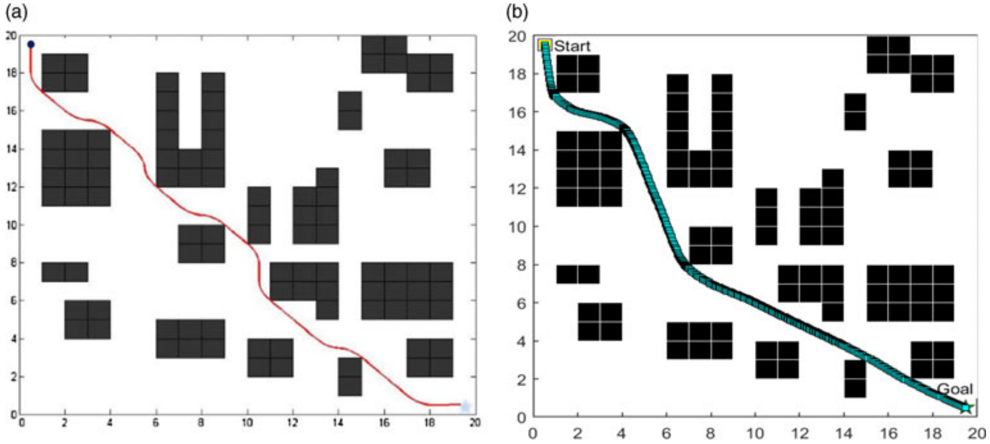

Figure 17. *(a) Trajectory achieved by Yen and Cheng [39] and (b) by the proposed improved butterfly optimization algorithm technique.*

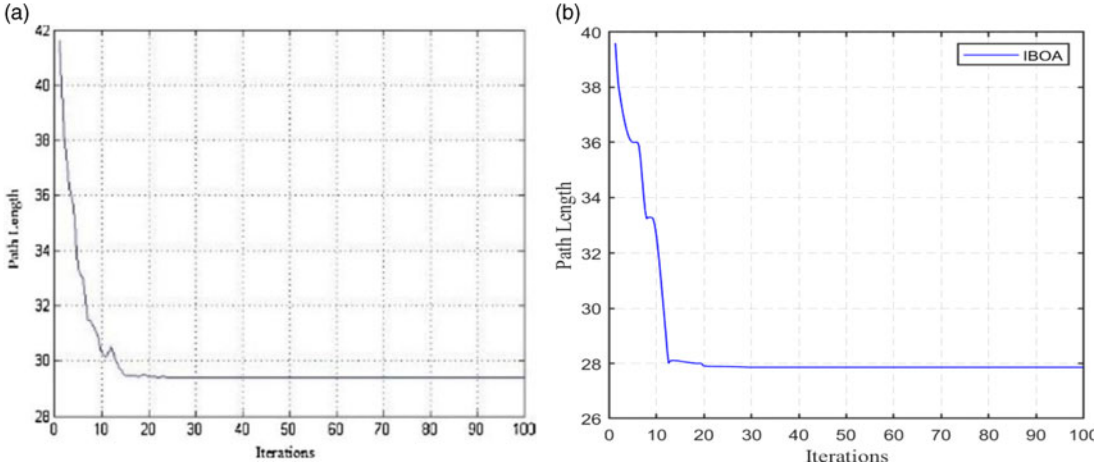

Figure 18. *(a) Convergence curve modeled by Yen and Cheng [39] and (b) by the proposed improved butterfly optimization algorithm technique.*

Author contributions. **Himansu Sekhar Dash:** Conceptualization, methodology, MATLAB simulation, Webots Simulation, experimental analysis, statistical analysis, paper original draft, editing, literature review, review and editing, and comparative analysis. **Dayal R. Parhi:** Supervision, conceptualization, methodology, MATLAB simulation, experimental analysis, statistical analysis, paper drafting, literature review, review and editing, and comparative analysis. **Manoj Kumar Muni:** Conceptualization, validation, and writing – review and editing. **Pinaki Das**: Literature review, writing – review and editing.

Financial support. This research received no specific grant from any funding agency, commercial, or not-for-profit sectors.

Competing interests. The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Ethical standards. Not applicable.

References

[1] A. Robotics (2012). Nao Documentation. http://www.aldebaran-robotics.com/documentation.

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

[3] Y. Ji, L. Ni, C. Zhao, C. Lei, Y. Du and W. Wang, “TriPField: A 3D potential field model and its applications to local path planning of autonomous vehicles,” *IEEE Trans. Intell. Transport. Syst.* **24**(3), 3541–3554 (2023).

[4] W. Li, C. Yang, Y. Jiang, X. Liu and C. Y. Su, “Motion planning for omnidirectional wheeled mobile robot by potential field method,” *J. Adv. Transport.* **2017**(1), 1–11 (2017).

[5] T. L. Lee and C. J. Wu, “Fuzzy motion planning of mobile robots in unknown environments,” *J. Intell. robot. Syst.* **37**(2), 177–191 (2003).

[6] X. Xia, T. Li, S. Sang, Y. Cheng, H. Ma, Q. Zhang and K. Yang, “Path planning for obstacle avoidance of robot arm based on improved potential field method,” *Sensors* **23**(7), 3754 (2023).

[7] G. Chen, N. Luo, D. Liu, Z. Zhao and C. Liang, “Path planning for manipulators based on an improved probabilistic roadmap method,” *Robot. Comput.-Integr. Manuf.* **72**, 102196 (2021).

[8] C. Mthabela, D. Withey and C. Kuchwa-Dube, “RRT based path planning for mobile robots on a 3D surface mesh, *2021 Southern African universities power engineering conference/robotics and mechatronics/pattern recognition association of south Africa (SAUPEC/RobMech/PRASA)*,” Potchefstroom, South Africa, IEEE (2021) pp. 1–6.

[9] R. A. Saeed, D. R. Recupero and P. Remagnino, “A boundary node method for path planning of mobile robots,” *Robot. Auton. Syst.* **123**, 103320 (2020).

[10] J. Cai, A. Du, X. Liang and S. Li, “Prediction-based path planning for safe and efficient human-robot collaboration in construction via deep reinforcement learning,” *J. Comput. Civil. Eng.* **37**(1), 04022046 (2023).

[11] Y. Yang, L. Juntao and P. Lingling, “Multi-robot path planning based on a deep reinforcement learning DQN algorithm,” *CAAI Trans. Intell. Technol.* **5**(3), 177–183 (2020).

[12] Q. Zhu, Y. Yan and Z. Xing. Robot path planning based on artificial potential field approach with simulated annealing, *Sixth international conference on intelligent systems design and applications*. Jian, China, IEEE (2006) Vol. 2, pp. 622–627.

[13] E. S. Low, P. Ong and K. C. Cheah, “Solving the optimal path planning of a mobile robot using improved Q-learning,” *Robot. Auton. Syst.* **115**, 143–161 (2019).

[14] J. Kennedy and R. Eberhart. Particle swarm optimization, *Proceedings of ICNN’95-international conference on neural networks*. Perth, WA, Australia, IEEE. (1995) Vol. 4, pp. 1942–1948.

[15] M. Brand, M. Masuda, N. Wehner and X. H. Yu. Ant colony optimization algorithm for robot path planning, *2010 international conference on computer design and applications*. Qinhuangdao, China, IEEE (2010) Vol. 3, pp. V3–436.

[16] J. M. Ahuactzin, E. G. Talbi, P. Bessiere and E. Mazer, “Using Genetic Algorithms for Robot Motion Planning,” **In:** *Geometric Reasoning for Perception and Action: Workshop Grenoble, France*, September 16–17, 1991 Selected Papers Springer, Berlin Heidelberg. (1993) pp. 84–93.

[17] K. C. Koh, H. R. Beom, J. S. Kim and H. S. Cho. A neural network-based navigation system for mobile robots, *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN’94)*. (1994) IEEE. Vol. 4, pp. 2709–2714.

[18] C. Liu, Z. Gao and W. Zhao. A new path planning method based on firefly algorithm, *2012 fifth international joint conference on computational sciences and optimization*. IEEE. (2012) pp. 775–778.

[19] A. M. Rao, K. Ramji and T. N. Kumar, “Intelligent navigation of mobile robot using grey wolf colony optimization,” *Materials Today: Proceedings* **5**(9), 19116–19125 (2018).

[20] W. Liu, B. Niu, H. Chen and Y. Zhu, “Robot path planning using bacterial foraging algorithm,” *J. Comput. Theor. Nanosci.* **10**(12), 2890–2896 (2013).

[21] P. K. Mohanty and D. R. Parhi. Cuckoo search algorithm for the mobile robot navigation, *Swarm, Evolutionary, and Memetic Computing: 4th International Conference, SEMCCO 2013*. Chennai, India, December 19-21, 2013, Proceedings, Part I 4, Springer International Publishing (2013) pp. 527–536.

[22] L. Lu and D. Gong. Robot path planning in unknown environments using particle swarm optimization, *2008 Fourth International Conference on Natural Computation*. Jinan, China, IEEE (2008) Vol. 4, pp. 422–426.

[23] Q. B. Zhang, P. Wang and Z. H. Chen, “An improved particle filter for mobile robot localization based on particle swarm optimization,” *Expert Syst. Appl.* **135**, 181–193 (2019).

[24] P. K. Das, H. S. Behera, S. Das, H. K. Tripathy, B. K. Panigrahi and S. K. Pradhan, “A hybrid improved PSO-DV algorithm for multi-robot path planning in a clutter environment,” *Neurocomputing* **207**, 735–753 (2016).

[25] F. K. Purian and E. Sadeghian. Mobile robots path planning using ant colony optimization and Fuzzy Logic algorithms in unknown dynamic environments, *2013 international conference on control, automation, robotics and embedded systems (CARE)*. Jabalpur, India, IEEE (2013) pp. 1–6.

[26] Z. Nie and H. Zhao. Research on robot path planning based on Dijkstra and Ant colony optimization, *2019 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*. Shanghai, China, IEEE (2019) pp. 222–226.

[27] H. Wang, J. Duan, M. Wang, J. Zhao and Z. Dong. Research on robot path planning based on fuzzy neural network algorithm, *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. Chongqing, China, IEEE (2018) pp. 1800–1803.

[28] N. Noguchi and H. Terao, "Path planning of an agricultural mobile robot by neural network and genetic algorithm," *Comput. Electron. Agric.* **18**(2-3), 187–204 (1997).

[29] J. Yu, Y. Su and Y. Liao, "The path planning of mobile robot by neural networks and hierarchical reinforcement learning," *Front. Neurorobot.* **14**, 63 (2020).

[30] Z. Garip, D. Karayel and M. Erhan Çimen, "A study on path planning optimization of mobile robots based on hybrid algorithm," *Concurr. Comput. Pract. Exp.* **34**(5), e6721 (2022).

[31] M. R. Panda, S. Dutta and S. Pradhan, "Hybridizing invasive weed optimization with firefly algorithm for multi-robot motion planning," *Arab. J. Sci. Eng.* **43**(8), 4029–4039 (2018).

[32] P. K. Mohanty and D. R. Parhi, "A new hybrid optimization algorithm for multiple mobile robots navigation based on the CS-ANFIS approach," *Memet. Comput.* **7**(4), 255–273 (2015).

[33] X. Yu, N. Jiang, X. Wang and M. Li, "A hybrid algorithm based on grey wolf optimizer and differential evolution for UAV path planning," *Expert. Syst. Appl.* **215**, 119327 (2023).

[34] S. P. Sahoo, B. Das, B. B. Pati, F. P. Garcia Marquez and I. Segovia Ramirez, "Hybrid path planning using a bionic-inspired optimization algorithm for autonomous underwater vehicles," *J. Marine Sci. Eng.* **11**(4), 761 (2023).

[35] S. Arora and S. Singh, "Butterfly optimization algorithm: A novel approach for global optimization," *Soft Comput.* **23**(3), 715–734 (2019).

[36] M. Yahya and M. P. Saka, "Construction site layout planning using multi-objective artificial bee colony algorithm with Levy flights," *Automat. Constr.* **38**, 14–29 (2014).

[37] C. Guimin, J. Jianyuan and H. Qi, "Study on the strategy of decreasing inertia weight in particle swarm optimization algorithm," *Journal-Xian Jiaotong University* **40**(1), 53 (2006).

[38] R. B. Wang, W. F. Wang, L. Xu, J. S. Pan and S. C. Chu, "An adaptive parallel arithmetic optimization algorithm for robot path planning," *J. Adv. Transport.* **2021**, 1–22 (2021).

[39] C. T. Yen and M. F. Cheng, "A study of fuzzy control with ant colony algorithm used in mobile robot for shortest path planning and obstacle avoidance," *Microsyst. Technol.* **24**(1), 125–135 (2018).

Cite this article: H. S. Dash, D. R. Parhi, M. K. Muni and P. Das (2025). "An improved butterfly optimization algorithm-based path navigation of humanoid robots in an unfamiliar setting", Robotica **43**, 2273–2303. https://doi.org/10.1017/S0263574725000712