Robotica: (2021) volume 39, pp. 123–136. © The Author(s) 2020. Published by Cambridge University Press.
doi:10.1017/S0263574720000235

Improved Motion Planning of Humanoid Robots Using Bacterial Foraging Optimization

Manoj Kumar Muni†*, Dayal R. Parhi† and Priyadarshi Biplab Kumar‡

†*Robotics Laboratory, Mechanical Engineering Department, National Institute of Technology Rourkela, Rourkela 769008, Odisha, India.*
E-mails: manoj1986nitr@gmail.com, dayaldoc@yahoo.com
‡*Mechanical Engineering Department, National Institute of Technology Hamirpur, Hamirpur 177005, Himachal Pradesh, India.*
E-mail: p.biplabkumar@gmail.com

(Accepted March 26, 2020. First published online: May 7, 2020)

SUMMARY
This paper emphasizes on Bacterial Foraging Optimization Algorithm for effective and efficient navigation of humanoid NAO, which uses the foraging quality of bacteria *Escherichia coli* for getting shortest path between two locations in minimum time. The Gaussian cost function assigned to both attractant and repellent profile of bacterium performs a major role in obtaining the best path between any two locations. Mathematical formulations have been performed to design the control architecture for humanoid navigation using the proposed methodology. The developed approach has been tested in a simulation platform, and the simulation results have been validated in an experimental platform. Here, motion planning for both single and multiple humanoid robots on a common platform has been performed by integrating a petri-net architecture for multiple humanoid navigation. Finally, the results obtained from both the platforms are compared in terms of suitable navigational parameters, and proper agreements have been observed with minimal amount of error limits.

KEYWORDS: Humanoid NAO; BFOA; Path planning; Petri-net.

1. Introduction

Humanoid robots, since their evolution, have attracted many researchers because of their human-like appearance. Although many researchers are working on humanoid robots, still path planning of humanoid robots from any source point to target location with moving obstacles is yet a challenging task. The main aspect in robots is proper navigation while moving from one point to other. There are so many paths available between two locations, but the goal is to find optimal path with collision-free environment. Generally, robots are used in industries to reduce human effort, so researchers have concentrated to develop humanoid robots which can look alike humans. In this regard, ALDEBARAN[1] group of France developed a small humanoid NAO. This developed NAO has features like twenty-five degrees of freedom and basic fundamental components such as ultrasonic and pressure sensors, cameras, actuators, emitters, receivers, and inbuilt NAOqi software that attract researchers to carry out their research to an enhanced stage. It has a height of 57 cm and weighing nearly 5 kg. Over the last few years, several investigations regarding motion planning for humanoid and other dorms of robots have been noticed and some of them can be highlighted over here.

* Corresponding author. E-mail: manoj1986nitr@gmail.com

Rath et al.[2,3] developed a fuzzy controller for navigation of a humanoid robot with avoiding obstacles. In this research, they provided the obstacle distances from robot and bearing angle as input to the controller and obtained respective velocity of robot to avoid obstacles. Patle et al.[4] developed and implemented firefly algorithm for path planning and avoidance of obstacles in mobile robots with static and dynamic environments. They demonstrated that the brightness of the firefly plays a key role while heading target, as variation in brightness attracts one firefly toward other. Rath et al.[5] proposed an artificial neural network controller for the effective path planning of a humanoid robot in both simulation and experimental environments. They used the sensor information like obstacle distances and location of target as input to the controller and obtained streaming angle as output from the controller. Rath et al.[6] presented genetic algorithm-based controller for path planning of a humanoid in complex environments with static and dynamic hurdles. They developed an objective function for path optimization and performed in both simulation and experimental platform. Dewang et al.[7] illustrated the adaptive particle swarm optimization for mobile robot navigation. In this research, the objective function was created in accordance with the distance between robot to target and robot to hurdle. Kumar et al.[8] proposed a hybridized regression adaptive ant colony optimization technique for the proper navigation of humanoid in less time. In this research, the obstacle distances sensed by the sensor are fed to the regression controller and the output from the regression controller is served to the ant colony controller to obtain the final output. Kumar et al.[9,10] presented regression and fuzzy logic controller technique for the navigation of humanoid from one location to other. Kumar et al.[11] proposed design and control of a manipulator arm for pick and place operations. Parhi et al.[12] combined adaptive swarm optimization method and adaptive ant colony optimization technique for better enhancement of humanoid toward path length. The inputs to the hybrid controller are front obstacle distance, left obstacle distance and right obstacle distance, and the corresponding output from the controller is turning angle. Paolillo et al.[13] presented a vision-based method for the humanoid robots, where the robots can travel the corridors effectively having junctions and curves. For the research analysis, they have used maze-like structure in corridors, T–junctions and curves. Abbas and Ali[14] developed enhanced bacteria foraging algorithm for path planning of autonomous mobile robots to find the optimal path in a two-dimensional workspace. They used the foraging quality of *Escherichia coli* (*E. coli*) bacteria to obtain the optimized path between two locations. Fakoor et al.[15] proposed fuzzy Markov decision method for path planning of humanoid robots in an unknown environment. Pandey and Parhi[16] focused on a hybrid technique and developed a hybridized rule-based fuzzy logic controller to avoid hurdles in complex environments during movement of humanoid. For the analysis, they provided three inputs to the controller and obtained one output from the controller. Parhi and Kumar[17] designed a new intelligent controller for navigational analysis of humanoid robots using a virtual target displacement strategy. Hornung et al.[18] developed localization method for navigation of humanoid robots in arbitrary complex platform with the help of onboard sensing. They used Monte-Carlo localization method and analyzed the humanoids six-dimensional torso pose in a three-dimensional model for staircase problem. Hossain and Ferdous[19] explored the Bacterial Foraging Optimization Algorithm (BFOA) for mobile robot navigation and obtained the shortest path in between two positions with avoiding the obstacles present in the arena. Faragasso et al.[20] presented a vision-based control approach for humanoids to navigate in office-like environment with maze-like corridors. Moharrei and Rad[21] used an augmented reality method and combined with a vision-based technique and developed the novel procedure for path planning of humanoid robots in complex environments. Feng et al.[22] applied the BFO algorithm to get the optimal path in between any source and target location. They also proposed Evolution-Strategy-Adaptive BFO for more accurate path while navigating between two locations. Zhao et al.[23] developed a neural network-based multiscale learning model to perform classification tasks.

Khatoon and Ibraheem[24] developed a hybridized navigation technique which comprises of two modules. In first step, they determined the rough optimal path toward target, and in the second step, potential function-based local approach is used for effective navigation decision. Nurmaini and Tutuko[25] used a weightless neural network and developed a new pattern recognition algorithm which takes proper control decision during mobile robot navigation. Khriji et al.[26] presented Q-learning technique for effective mobile robot navigation. This technique works with fuzzy analysis where the design of individual behavior is done and prior knowledge is fed to the Q-learning method. Parasuraman et al.[27] implemented modified fuzzy associative memory for robot navigation. This technique develops the rule base for robot path planning. Gueaieb et al.[28] proposed an innovative radio-frequency identification technique for mobile robot navigation in a prior unknown environment without using the vision-based method and creating a roadmap of the workspace. Abiyev et al.[29] presented a new type-2 fuzzy system for mobile robot navigation cluttered with hurdles in uncertain environment. The developed technique provides a relation between the inputs such as current angle and distance signals with output robot turn angle. Khairunizam et al.[30] carried out fuzzy membership function controller with inputs as motion information from the movements of two-wheeled robots and obtained the forward velocity of mobile robot as output while traversing from source to target. Kowalczyk[31] proposed a navigation function controller which generates the motion direction of mobile robot and determines the velocity of robot during movement or turning while heading toward the target. Shayestegan and Marbaban[32] developed a new Braitenberg strategy for smooth navigation of mobile robot. They developed a switching command strategy for efficient and effective motion path of mobile robot. Parhi and Chhotray[33] developed a new novel DAYANI arc contour intelligent technique for hurdle for optimal path navigation of two-wheeled mobile robot in complex environment. Juang et al.[34] presented AF-CACPSO technique for path planning of mobile robot in unknown terrain. This technique primarily works on the principles such as obstacle boundary following or target seeking. Chou and Juang[35] proposed an approach for navigational purpose which consists of three segments such as hurdle avoidance behavior, target seeking behavior and a behavior supervisor. They developed a pareto fuzzy controller with continuous ant colony optimization technique for obstacle avoidance behavior and a hybrid Proportional Integral Derivative controller for seeking target Armah et al.[36] presented an effective navigation architecture for mobile robots which works on the principles of go-to-goal, avoid obstacle and follow wall. Zolghadr and Cai[37] used extended Kalman filter with triangular, circular, elliptical and sinusoidal path to extract the correct data for navigation of two-wheeled robot. Gutmann et al.[38] presented A* algorithm for real-time navigation of humanoid robot in static and dynamic environment. Kuffner et al.[39] proposed a foot step planning method for safe navigation of biped robots in moving obstacle prone environment. Koh and Cho[40] presented a bang-bang control technique to obtain the driving velocity of mobile robot while heading toward goal.

From the extensive survey of literature, it can be observed that bacterial foraging optimization being an effective problem-solving approach has a very limited application toward navigational analysis of humanoid robots. Therefore, the current research is aimed at design and development of a novel BFOA-based motion planning strategy for smooth and hassle-free navigation of single as well as multiple humanoid robots in complex arenas. Here, a petri-net model has been combined with the basic BFOA controller to negotiate dynamic obstacles in the arena.

2. General Overview of BFOA

BFOA is one of the nature-inspired methods to discover the best solution of any problem. This optimization technique was proposed by Prof. K. M. Passino in 2002,[41] which works on the social foraging behavior of *E. coli* bacteria extant in human intestine. In this method, bacteria hunt for food or nutrient-rich region in a way to obtain maximum energy per time. This activity provides more time to bacteria for performing major deeds as well as gives nutrient-rich bases to work. Another aspect in this approach is the communication between the bacteria by sending signals. During searching, the real bacteria perform two simple actions with the help of tensile flagella. They are namely swim and tumble.

2.1. Basic of BFOA

This algorithm works on the principle of calculated values of cost function. In this process, the bacteria follow the steps where it finds low-cost function value or high fitness. When the steps continue, it leads to better fitness with low-cost function. The location of the bacteria is optimized and fragmented in the permissible range where each fragmented factor represents a point in the space coordinates. At each location, the cost function is determined and with this value, the movement of bacteria takes place according to the decreased direction of cost value. This finally keeps the bacteria to a location with utmost fitness.

The algorithm comprises four steps; they are as follows.

(i) Chemotaxis; (ii) Swarming; (iii) Reproduction; (iv) Elimination and Dispersal

2.1.1. Chemotaxis. This operation is accomplished by swimming and tumbling. Swim refers to a unit length in the same direction, whereas tumble means a unit walk in the different direction as compared to the previous direction. With the nutrient-rich and noxious-free environment which is favorable for the bacterium, the bacteria will swim in the same path it is following. This can be shown by Fig. 1.

The bacterium will change the swimming direction to tumble when an unfavorable condition arises. It can be shown by Fig. 2.

Let α (i, j, k, l) represents the bacterium at jth chemotactic, kth reproduction and lth elimination-dispersal step. $RT(i)$ shows chemotactic step size during each run or tumble operation. The movement of the ith bacterium can be expressed as:

$$\alpha(i, j+1, k, l) = \alpha(i, j, k, l) + RT(i) \times \mu(i) \tag{1}$$

$$\mu\ (i) = \frac{\Phi\ (i)}{\sqrt{\Phi^T\ (i)\Phi\ (i)}}$$

where
μ (i) = Unit vector represents for the swimming direction after a tumble

$\Phi\,(i)$ = Randomly fashioned vector with same dimension

Suppose C (i, j, k, l) represents the cost at the location of the bacteria α (i, j, k, l). If at α $(i, j+1, k, l)$, the cost C $(i, j+1, k, l)$ is smaller than the cost C (i, j, k, l) at α (i, j, k, l), then the bacterium will move to a step size of $RT(i)$ in the same direction. Otherwise, the bacterium will change its direction by tumbling with step size of $RT(i)$ in order to obtain better nutrient-rich and noxious-free environment.

2.1.2. Swarming. *E. coli* bacterium has a decision-making tool, a quality of signaling other bacteria to swarm in the same way by using attractant profile. This step is known as cell to cell mechanism which also releases chemical repellent to inform other bacteria to retain safe distance from it. The attraction and repulsion effect in BFOA can be expressed as:

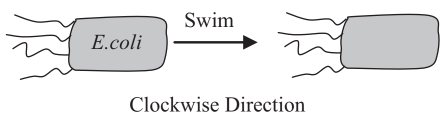

Fig. 1. Swim operation in BFOA.

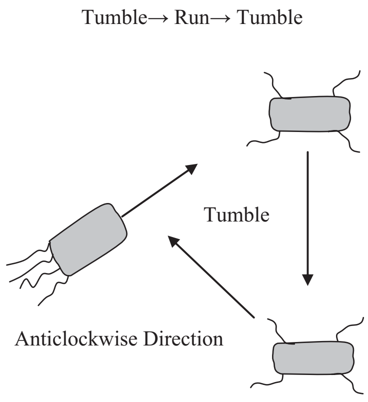

Fig. 2. Tumble operation in BFOA.

https://doi.org/10.1017/S0263574720000235 Published online by Cambridge University Press

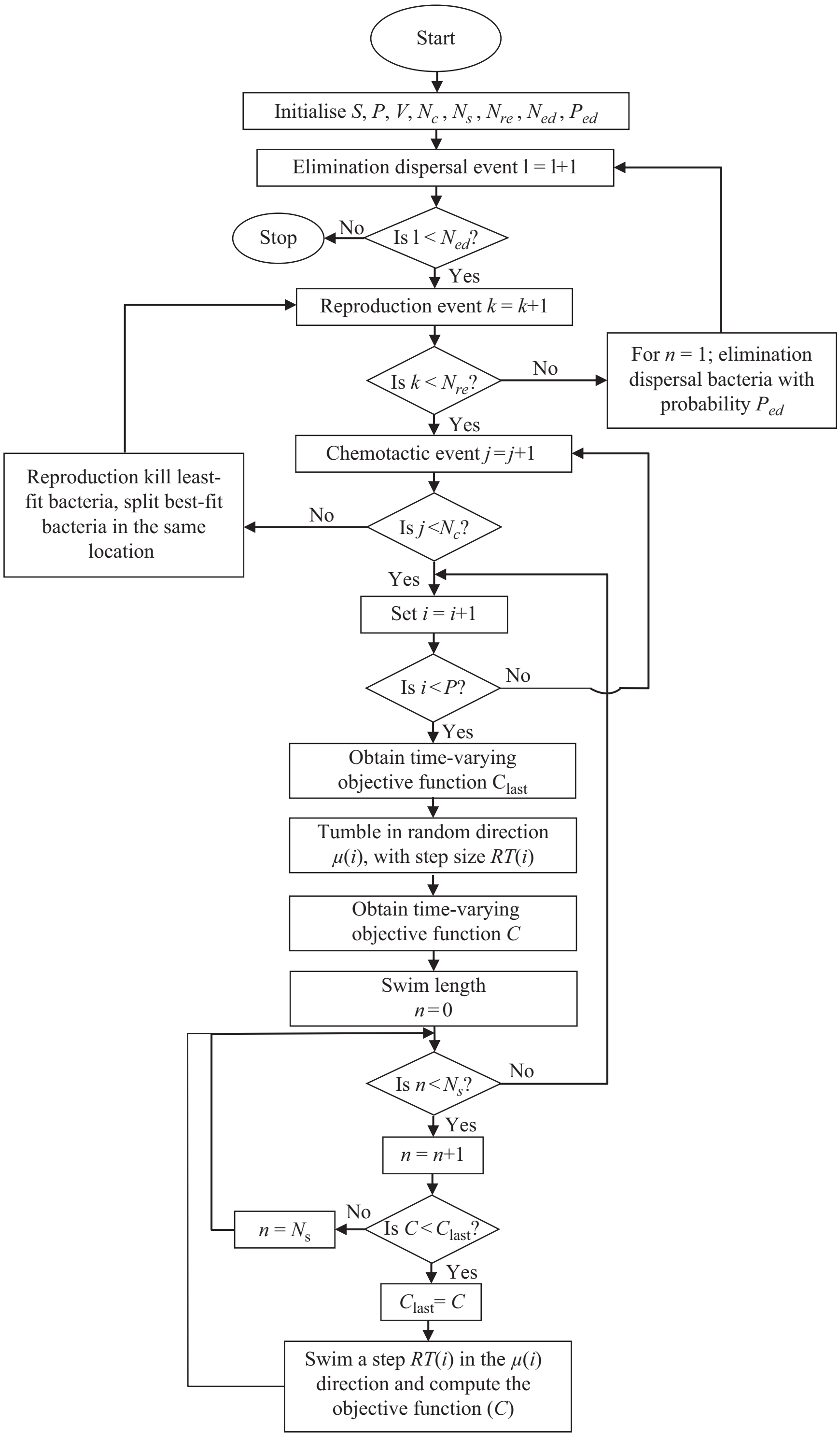

Fig. 3. Flowchart of BFOA.

$$C_{cc}\left(\alpha, R\left(j, k, l\right)\right) = \sum_{i=1}^{P} C_{cc}\left(i\right)\left(\alpha, \alpha\left(i, j, k, l\right)\right) = \sum_{i=1}^{P}\left[-d_{attr} e^{\left(-w_{attr}\sum_{n=1}^{V}(\alpha_n - \alpha_n(i))^2\right)}\right] + \sum_{i=1}^{P}\left[-h_{repel} e^{\left(-w_{repel}\sum_{n=1}^{V}(\alpha_n - \alpha_n(i))^2\right)}\right] \tag{2}$$

https://doi.org/10.1017/S0263574720000235 Published online by Cambridge University Press

Where $C_{cc}(\alpha, R(j, k, l))$ represents the time-varying objective function when added to actual objective function. P denotes the total number of bacteria present in search space. V represents the total number of variables present in search space $\alpha = [\alpha_1, \alpha_2, ...\alpha_V]^T$. $\alpha_n(i)$ signifies nth component of ith bacterium location. Different coefficients used in the above equation are d_{attr}, W_{attr}, h_{repel} and W_{repel}. Care should be taken properly while selecting the coefficients.

2.1.3. Reproduction. This is an important step of BFOA. If total number of bacteria decreases while performing the algorithm, then it is not possible to achieve the desired goal. So in order to maintain bacteria count constant, reproduction is much needed. In this stage, the population is arranged in ascending order of fitness value of the bacteria. After arranging only 50% of best fit, bacteria continue to reproduce (split into two bacteria) so that the population remains constant and other least fit bacteria dies.

2.1.4. Elimination and dispersal. There is an effect on the lives of bacteria takes place, when sudden changes in the local condition changes. This may happen either by local rise in temperature or by sudden water flow. In both the cases, the bacteria may die or a collection of bacteria may be dispersed into new position. With a probability of P_{ed}, the elimination of bacteria takes place and fresh additional bacteria are randomly placed in the search space.

2.2. Bacterial foraging optimization algorithm

In the very beginning, we have to initialize the parameters such as,

S = Dimension of search space
V = Total number of variables present in search space
P = Total number of bacteria in the population
N_c = Number of Chemotactic events
N_s = Swimming length
N_{re} = Number of reproduction events
N_{ed} = Number of elimination-dispersal steps
P_{ed} = Probability of elimination-dispersal
$RT(i)$ = Step size in random direction specified by tumble

Figure 3 represents the flowchart of the BFOA used in the current work.

3. Navigation of Humanoid NAO using BFOA

The flowchart shown in Fig. 3 activates when NAO senses any obstacle while navigating from one position to other. First sensor helps to collect the information of environment having hurdles, and then humanoid performs its goal of reaching target effectively without colliding the obstacles.

3.1. Decision strategy for getting optimal bacterium

The humanoid's objective is to move from source to target in less time with shortest path, and avoiding obstacles present nearby. In this algorithm, if the NAO senses any hurdle in its path, then the BFOA activates, which generates a group of bacterium randomly near to it. With radius same as step size, the next location of bacterium can be expressed as:

$$\Delta_L (t + \delta t) = \Delta_L (t) + RT (i) * \mu (i) \tag{3}$$

Where $\Delta_L(t)$ = Location of each bacterium at time t.

When time increases such as from t to $t + \delta t$, the bacterium heads toward the target by searching optimal path and avoiding the hurdles. Using Eq. (3), the bacterium which gets the best-fit path for the upcoming location is selected and the NAO moves to the new location. This process continues until the NAO reaches the target. Two factors are responsible while selecting the best bacterium in order to get optimized path. They are (i) distance between NAO and hurdle and (ii) distance of the nearest hurdle with NAO's current location. When these two factors combine, an attractant–repellent outline is developed which attracts the NAO toward the target having global minima and repels the NAO from the nearest hurdles.

The main objective of BFOA is to keep NAO away from hurdles or a safe distance from obstacles. When the NAO senses any hurdle, then repulsive Gaussian cost function is assigned to C_{hurdle} which can be expressed as

$$C_{hurdle} = h_{hurdle} * \left[\frac{1}{e^{\left(w_{hurdle} * \left(\| \Delta_i(t) - L_{hurdle}(t) \|^2\right)\right)}} \right] \quad (4)$$

where h_{hurdle} = height coefficient of repellent

w_{hurdle} = width coefficient of repellent
L_{hurdle} = location of hurdle sensed by the NAO through sensors
$\| \Delta_i(t) - L_{hurdle}(t) \|^2$ = distance between the NAO's current location and the hurdle nearby

Eq. (4) has only value, when the NAO senses any hurdle in its sensing range. So, the cost functions C_{hurdle} can be given by:

$$C_{hurdle} = \begin{cases} h_{hurdle} * \left[\frac{1}{e^{\left(w_{hurdle} * \left(|\Delta_i(t) - L_{hurdle}(t)|^2\right)\right)}} \right] & \text{when} |\Delta_i(t) - L_{hurdle}(t)|^2 \le \nu_{NAO} \\ 0 & \text{when} |\Delta_i(t) - L_{hurdle}(t)|^2 \ge \nu_{NAO} \end{cases} \quad (5)$$

Where ν_{NAO} is the sensing range of NAO. Likewise the repellent Gaussian cost function, an attractant Gaussian cost function is assigned to the target location. The target cost function can be shown by:

$$C_{target} = h_{target} * \left[\frac{1}{e^{\left(w_{target} * \left(\| \Delta_i(t) - L_{target} \|^2\right)\right)}} \right] \quad (6)$$

Where

h_{target} = height coefficient of attractant
w_{target} = width coefficient of attractant
L_{target} = predefined target location
$\left\| \Delta_i(t) - L_{target} \right\|^2$ = Euclidean distance from NAO's current position and predefined target location

The general cost function which involves to keep the hurdles away from NAO and Target nearby to NAO can be represented as:

$$C_{total} = C_{hurdle} + C_{target} \quad (7)$$

Eq. (7) is used to find out the cost function of each bacterium. The bacterium is arranged in a manner having low-cost function value and high fitness value. The NAO moves to the location of bacterium which is having low-cost function value. By continuing above methods with reproduction, and elimination-dispersal steps, the humanoid robot heads toward the target location in shortest path and minimum time.

3.2. Decision strategy for minimizing distance error

A decision controller is developed to obtain the best bacterium among the randomly generated bacterium around the NAO within the small time δt. The distance error of bacterium to target location $de_i^{dist}(t + \delta t)$ and the cost function error $ce_i^C(t + \delta t)$ help in obtaining best bacterium for the desired goal. The distance error and cost function error can be expressed as:

$$de_i^{dist}(t + \delta t) = d_i(t + \delta t) - d_i(t)$$
$$ce_i^C(t + \delta t) = C_{total}(\Delta_i(t + \delta t)) - C_{total}(\Delta_i(t))$$

https://doi.org/10.1017/S0263574720000235 Published online by Cambridge University Press

Where

$d_i\,(t+\delta t) \;=\; \left\|\Delta_i\,(t) - L_{target}\right\|^2$ = distance of bacterium B to the target location at time $t+\delta t$

$d_i\,(t) = \left\|\Delta_i\,(t) - L_{target}\right\|^2$ = distance of bacterium B to the target location at time t

The distance error and cost function error of each bacterium are determined at time $t+\delta t$ and arranged in ascending manner such that the bacterium with high error is at last. The bacterium with negative cost function error is selected in accordance with minimum distance error. If the cost function error of bacterium is greater than zero, then it signifies the hurdle prone region. So, the bacterium with high negative distance error is chosen as best bacterium and NAO takes the location of best bacterium in the consequent steps.

4. Performance Analysis of Developed BFOA Controller

To examine the efficiency of the developed controller in terms of path span and time spent, the simulation and experimental platforms are created with dynamic obstacles using NAO H25 V4. The motion planning of single and multiple humanoids is performed in both simulation and experimental scenario.

4.1. Motion planning of a single humanoid

For the simulation work, V-REP software is considered as a navigational tool for its improved motion planning approach with obstacle avoidance character. The coding is written in Python programming language and fed to the V-REP software for navigational purpose. Multiple static obstacles with different sizes are randomly placed at arbitrary points of simulation platform with definite source and goal position. Fig. 4 represents the path followed by the NAO during the simulation.

The experimental platform is created as per the simulation scenario with obstacles to authenticate the developed controller. In experimental motion planning execution, the developed controller activates during the path travel and feds the code written in MATLAB to NAO for progress toward the goal. Fig. 5 represents the real-time path travelled by the humanoid in laboratory conditions. Tables I and II show the comparison of results.

The results of simulation and experimental arena are compared in terms of navigational parameters and are presented in Tables I and II.

Table I. Comparison of path span.

Sl. no.	Path span in simulation (cm)	Path span in experiment (cm)	% Error
1	290.5	310.5	6.44
2	291.67	312.6	6.7
3	290.53	312.2	6.94
4	291.46	311.3	6.37
5	291.03	311.7	6.63
Average	291.04	311.66	6.62

Table II. Comparison of time spent.

Sl. no.	Time spent in simulation (s)	Time spent in real-time (s)	% Error
1	40.46	43.32	6.6
2	40.72	43.56	6.52
3	40.35	43.03	6.23
4	40.83	43.45	6.03
5	40.11	43.52	7.84
Average	40.49	43.38	6.64

https://doi.org/10.1017/S0263574720000235 Published online by Cambridge University Press

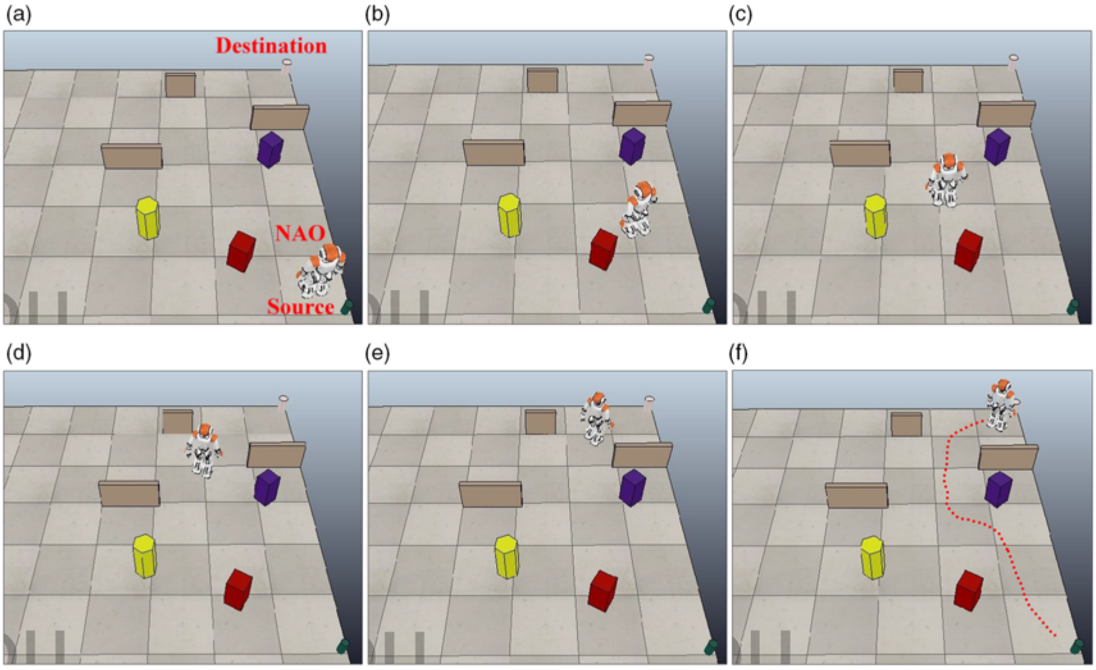

Fig. 4. Simulation of a single NAO.

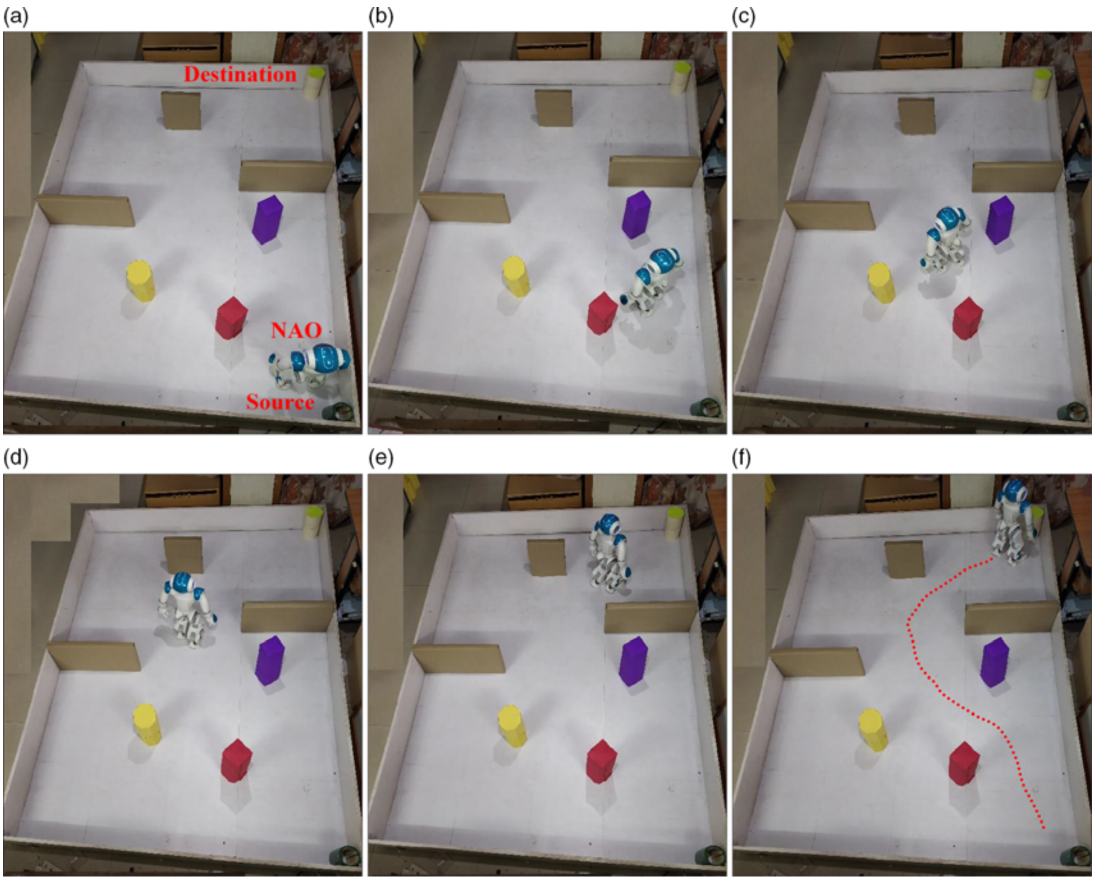

Fig. 5. Experiment of a Single NAO.

https://doi.org/10.1017/S0263574720000235 Published online by Cambridge University Press

It can be observed that the simulation and experimental results are in well agreement with each other. The reason for observation of higher values of navigational parameters in experimental results compared to simulation counterparts can be justified by the presence of factors like slippage effect, data transmission loss and frictional losses in the experimental platform which are ideal in case of the simulation platform.

4.2. Motion planning of multiple humanoids

Navigation of multiple humanoids is very much similar to that of the article represented in Section 4.1, but some differences can be found in number of humanoid NAO and number of source and target positions. Due to the presence of dynamic obstacles in the path, a petri-net control scheme is associated with the developed BFOA controller to avoid conflicts that may arise during the navigation.

4.2.1. Petri-net control scheme. In the motion planning of multiple NAOs in dynamic environment with moving obstacles, there are some conflicts arise among the humanoids when they sense a conjoint obstacle during navigation. The developed controller aims to detect obstacles and smooth navigation toward reaching the target by single and multiple humanoids in a common scenario. When two or more humanoid detects same obstacle in their path, then the developed controller fails to provide the priority of motion planning. To overcome the problem, a petri-net controller is integrated with the developed scheme, which avoids the inter collision among the NAOs by providing the information of navigational priority. Figure 6 shows a petri-net model with different stages used in the current analysis.

There are six stages in the model, and each stage is described as follows:

- Stage 1: In this stage, all the robots are placed randomly in the platform and are not having any prior idea about the location of other humanoids. All the humanoids approach toward goal, when the start command activates.
- Stage 2: This stage is related to target seeking behavior, where all the NAOs move toward the goal point and they may sense obstacles in their path.
- Stage 3: This stage represents the obstacle detection stage.
- Stage 4: In this stage, the priority of motion is given to the humanoid which is nearer to the goal and moves forward, while the others act as an obstacle in their respective locations.
- Stage 5: In this stage, the humanoids move toward goal after finding null conflicting conditions.
- Stage 6: This stage represents the waiting stage where the humanoids having interaction with stage 3 NAOs behave as an obstacle and wait till the first set of humanoid travels ahead. After this, the waited humanoids perform their motion planning approach from stage 2.

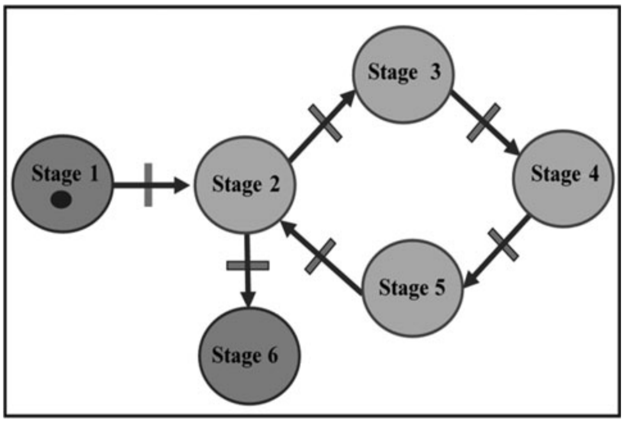

Fig. 6. Proposed petri-net model.

https://doi.org/10.1017/S0263574720000235 Published online by Cambridge University Press

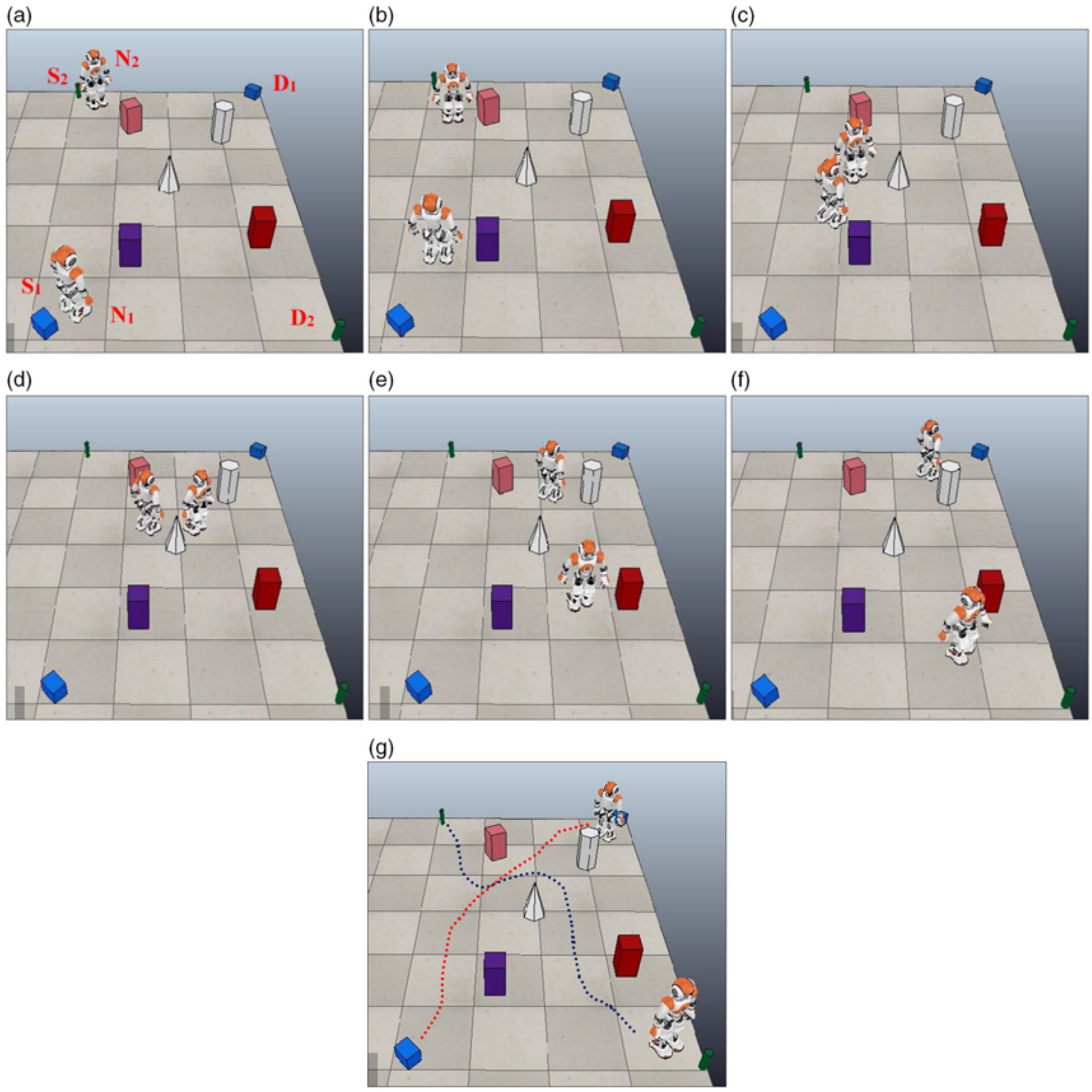

Fig. 7. Simulation of multiple NAOs.

The developed controller activates whenever it finds any obstacle in its path by any humanoid. Figs. 7 and 8 show the outcomes of simulation navigation and experimental motion planning. Tables III and IV represent the comparison of navigational parameters.

The results obtained from the simulation and experimental analysis are found to be satisfactory and are well within the acceptable limit.

5. Conclusions

In this paper, BFOA as a potential navigational approach for complex motion planning of humanoid robots has been thoroughly discussed. The mathematical formulations regarding the navigational approach have been analyzed. NAO humanoid robots have been used as the humanoid platform for the navigational analysis. The developed BFOA methodology has been implemented in V-REP simulation platform, and the simulation results have been validated through an experimental platform. Here, the navigational analysis has been performed on single as well as multiple humanoid models. To navigate multiple humanoids on a common platform, a petri-net architecture has been combined with the proposed navigational strategy. Finally, the results obtained from both the platforms have been compared in terms of selected navigational parameters, and suitable agreements have been observed.

Fig. 8. Experiment of multiple NAOs.

Table III. Comparison of path span.

	Simulation arena		**Real-time arena**		**% Error**	
	Path span (cm)					
Sl. no.	N_1	N_2	N_1	N_2	N_1	N_2
1	315.46	303.42	337.2	325.1	6.45	6.67
2	315.87	303.81	337.8	325.2	6.49	6.58
3	315.76	304.26	338.4	328.4	6.69	7.35
4	314.93	305.99	339.2	326.8	7.16	6.37
5	316.86	304.83	340	326.3	6.81	6.58
Average	315.78	304.46	338.52	326.36	6.72	6.71

https://doi.org/10.1017/S0263574720000235 Published online by Cambridge University Press

Table IV. Comparison of time spent.

	Simulation arena		Experimental arena		% Error	
	Time spent (s)					
Sl. no.	N_1	N_2	N_1	N_2	N_1	N_2
1	43.21	41.36	46.24	44.4	6.55	6.85
2	43.23	41.47	46.47	44.55	6.97	6.91
3	43.15	41.58	46.33	44.67	6.86	6.92
4	43.76	41.73	46.18	44.86	5.24	6.98
5	43.55	41.54	46.92	44.64	7.18	6.94
Average	43.38	41.54	46.43	44.62	6.56	6.92

References

1. A. Robotics, “Nao Documentation” v1. 14.2.
2. A. K. Rath, D. R. Parhi, H. C. Das, M. K. Muni and P. B. Kumar, “Analysis and use of fuzzy intelligent technique for navigation of humanoid robot in obstacle prone zone,” *Defence Technol.* **14**(6), 677–682 (2018).
3. A. K. Rath, D. R. Parhi, H. C. Das, P. B. Kumar, M. K. Muni and K. Salony, “Path optimization for navigation of a humanoid robot using hybridized fuzzy-genetic algorithm,” *Int. J. Intell. Unmanned Syst.* **7**(3), 112–119 (2019).
4. B. K. Patle, A. Pandey, A. Jagadeesh and D. R. Parhi, “Path planning in uncertain environment by using firefly algorithm,” *Defence Technol.* **14**(6), 691–701 (2018).
5. A. K. Rath, H. C. Das, D. R. Parhi and P. B. Kumar, “Application of artificial neural network for control and navigation of humanoid robot,” *J. Mech. Eng. Sci.* **12**(2), 3529–3538 (2018).
6. A. K. Rath, D. R. Parhi, H. C. Das and P. B. Kumar, “Behaviour based navigational control of humanoid robot using genetic algorithm technique in cluttered environment,” *Measur. Control A* **91**(1), 32–36 (2018).
7. H. S. Dewang, P. K. Mohanty and S. Kundu, “A robust path planning for mobile robot using smart particle swarm optimization,” *Procedia Comput. Sci.* **133**, 290–297 (2018).
8. P. B. Kumar, C. Sahu and D. R. Parhi, “A hybridized regression-adaptive ant colony optimization approach for navigation of humanoids in a cluttered environment,” *Appl. Soft Comput.* **68**, 565–585 (2018).
9. P. B. Kumar, S. Mohapatra and D. R. Parhi, “An intelligent navigation of humanoid NAO in the light of classical approach and computational intelligence,” *Comput. Animat. Virtual Worlds* **30**(2), e1858.
10. P. B. Kumar, N. K. Verma, D. R. Parhi and D. Priyadarshi, “Design and control of a 7 DOF redundant manipulator arm,” *Australian J. Mech. Eng.,* 1–12 (2019). doi:10.1080/14484846.2019.1656354.
11. P. B. Kumar, M. Sethy and D. R. Parhi, “An intelligent computer vision integrated regression based navigation approach for humanoids in a cluttered environment,” *Multimedia Tools Appl.* **78**(9), 11463–11486 (2019).
12. D. R. Parhi, C. Sahu and P. B. Kumar, “Navigation of multiple humanoid robots using hybrid adaptive swarm-adaptive ant colony optimisation technique,” *Comput. Anim. Virtual Worlds* **29**(2), e1802.
13. A. Paolillo, A. Faragasso, G. Oriolo and M. Vendittelli, “Vision-based maze navigation for humanoid robots,” *Autonomous Robots* **41**(2), 293–309 (2017).
14. N. H. Abbas and F. M. Ali, “Path planning of an autonomous mobile robot using enhanced bacterial foraging optimization algorithm,” *Al-Khwarizmi Eng. J.* **12**(4), 26–35 (2016).
15. M. Fakoor, A. Kosari and M. Jafarzadeh, “Humanoid robot path planning with fuzzy Markov decision processes,” *J. Appl. Res. Technol.* **14**(5), 300–310 (2016).
16. A. Pandey and D. R. Parhi, “MATLAB simulation for mobile robot navigation with hurdles in cluttered environment using minimum rule based fuzzy logic controller,” *Procedia Technol.* **14**, 28–34 (2014).
17. D. R. Parhi and P. B. Kumar, “Smart navigation of humanoid robots using DAYKUN-BIP virtual target displacement and petri-net strategy,” *Robotica* **37**(4), 626–640 (2019).
18. A. Hornung, S. Oßwald, D. Maier and M. Bennewitz, “Monte Carlo localization for humanoid robot navigation in complex indoor environments,” *Int. J. Humanoid Robotics* **11**(02), 1441002 (2014).
19. M. A. Hossain and I. Ferdous, “Autonomous robot path planning in dynamic environment using a new optimization technique inspired by bacterial foraging technique,” *Robotics Auton. Syst.* **64**, 137–141 (2015).
20. A. Faragasso, G. Oriolo, A. Paolillo and M. Vendittelli, “Vision-Based Corridor Navigation for Humanoid Robots,” 2013 IEEE International Conference on Robotics and Automation (ICRA), Karlsruhe, Germany (2013) pp. 3190–3195.
21. O. Mohareri and A. B. Rad, “Autonomous Humanoid Robot Navigation Using Augmented Reality Technique,” Proceedings of the 2011 IEEE International Conference on Mechatronics, Istanbul, Turkey (2011) pp. 463–468.
22. X. Feng, Y. He, H. Yang and Y. Juan, “Self-adaptive bacterial foraging optimization algorithm based on evolution strategies” *Rev. Téc. Ing. Univ. Zulia* **39**(8), 350–358 (2016).
23. Y. Zhao, J. Pei and H. Chen, “Multi-layer radial basis function neural network based on multi-scale kernel learning,” *Appl. Soft Comput.* **82**, 105541 (2019).

https://doi.org/10.1017/S0263574720000235 Published online by Cambridge University Press

24. S. Khatoon and I. Ibraheem, "Autonomous mobile robot navigation by combining local and global techniques" *Int. J. Comput. Appl.* **37**(3), 1–10 (2012).
25. S. Nurmaini and B. Tutuko, "A new classification technique in mobile robot navigation" *TELKOMNIKA* **9**(3), 453–464 (2013).
26. L. Khriji, F. Touati, K. Benhmed and A. Al-Yahmedi, "Mobile robot navigation based on Q-learning technique" *Int. J. Adv. Robotic Syst.* **8**(1), 4 (2011).
27. S. Parasuraman, B. Shirinzadeh and V. Ganapathy, "Sensors fusion technique for mobile robot navigation using fuzzy logic control system" *Mobile Robots Navig.*, 85–106 (2010).
28. W. Gueaieb and M. S. Miah, "An intelligent mobile robot navigation technique using RFID technology" *IEEE Tran. Inst. Meas* **57**(9), 1908–1917 (2008).
29. R. H. Abiyev, B. Erin and A. Denker, "Navigation of Mobile Robot Using Type-2 Fuzzy System" International Conference on Intelligent Computing (ICIC), Liverpool, United Kingdom (2017) pp. 15–26.
30. W. A. N. Khairunizam, R. M. Nor, M. N. Ayob, D. Hazry, A. B. Shahriman and M. Zuradzman, "Mobile robot navigation by using fuzzy information of moving two-wheeled motion" *Int. J. Mech. Mechatron. Eng. IJMME-IJENS* **13**(4), 34–39 (2013).
31. W. Kowalczyk, "Rapid navigation function control for two-wheeled mobile robots" *J. Intell. Robotic Syst.* **93**(3–4), 1–11 (2018).
32. M. Shayestegan and M. H. Marhaban, "A braitenberg approach to mobile robot navigation in unknown environments" International Conference on Intelligent Robotics, Automation, and Manufacturing (IRAM), Kuala Lumpur, Malaysia (2012) pp. 75–93.
33. D. R. Parhi and A. Chhotray, "Development and analysis of DAYANI arc contour intelligent technique for navigation of two-wheeled mobile robot" *Ind. Robot Int. J.* **45**(5), 688–702 (2018).
34. C. F. Juang, M. G. Lai and W. T. Zeng, "Evolutionary fuzzy control and navigation for two wheeled robots cooperatively carrying an object in unknown environments" *IEEE Trans. Cybe.* **45**(9), 1731–1743 (2015).
35. C. Y. Chou and C. F. Juang, "Navigation of an autonomous wheeled robot in unknown environments based on evolutionary fuzzy control" *Inventions* **3**(1), 3 (2018).
36. S. Armah, S. Yi and T. Abu-Lebdeh, "Implementation of autonomous navigation algorithms on two wheeled ground mobile robot" *Am. J. Eng. Appl. Sci.* **7**(1), 149–164 (2014).
37. J. Zolghadr and Y. Cai, "Locating a two-wheeled robot using extended Kalman filter" *Tehnièki vjesnik* **22**(6), 1481–1488 (2015).
38. J. S. Gutmann, M. Fukuchi and M. Fujita, "Real-Time Path Planning for Humanoid Robot Navigation" Nineteenth International Joint Conference on Artificial Intelligence (IJCAI), Edinburgh, Scotland (2005) pp. 1232–1237.
39. J. J. Kuffner, K. Nishiwaki, S. Kagami, M. Inaba and H. Inoue, "Footstep Planning Among Obstacles for Biped Robots" Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2001) pp. 500–505.
40. K. C. Koh and H. S. Cho, "A smooth path tracking algorithm for wheeled mobile robots with dynamic constraints" *J. Intell. Robotic Syst.* **24**(4), 367–385 (1999).
41. K.M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control" *IEEE Control Syst. Mag.* **22**(3), 52–67 (2002).