NANOBIOTECHNOLOGY REPORTS
ISSN:2635-1676E-ISSN:2635-1684

DOI: 10.5281/zenodo.7479138

AN OPTIMAL MOTION PLANNING OF ROBOTICS THROUGH REACTIVE APPROACH AND GENETIC ALGORITHM

[1]Rajnesh Kumar Singh,[2]Neelu Jain

1 Research scholar, Punjab Engineering College, Chandigarh and Assistant Professor, ABES Engineering College, Ghaziabad, Uttar Pradesh, India

2Professor,Punjab Engineering College, Chandigarh

Abstract

This paper shows the performance of robotic motion planning using a reactive approach and the Genetics Algorithm (GAs). Naturally, Robots have to face a lot of obstacles in their path. To avoid the collision with obstacles, robots change their path by controlling the velocity of both the left side and the right side wheel. The controlling of wheel velocity has been assessed by the reactive approach. The major task for the robot has to minimize the time taken to travel from one coordinate to another. The topology of estimating the time for robot motion has been assessed with a genetic algorithm. It is observed that robots takes minimum time with genetic algorithm in comparison to other existing techniques.

Keywords: Robotics; GAs; reactive; time; Obstacle

1. Introduction

Robotics is an engineering and technology branch that includes electronics, mechanical and computer science engineering, and so on. This section includes the design, development, utilization of robot control, sensory information, and processing of information. Several innovations shortly will substitute living things and natural processes. Such robots are meant to be used in any intent, but they are used in hazardous areas such as bomb identification, multiple bomb deletion, etc. Robots may take any shape, but the human look has been given to several of them. Robots have a physical structure. They are kept by their body's framework and are driven by the mechanical parts Robots would just be a computer program without presence. Another brain name in robots is On-Board Control Device. It receives information that uses this robot and gives a signal as the output. This will be only a remote-controlled computer with this position sensor robot knowing what to do else. The use of such sensors in robots is to collect information from the world and give this to the Brain. Practically, there is circuitry in such sensors that create the voltage in themselves. The machines push and the parts are called actuators with the aid of these robots.

When the robot moves to find its target, it has to face a lot of hurdles or obstacles due to which the robot deviates from its natural path and mission didn’t accomplish. The duty assigned to the robot with a certain set of training rules and guidance for avoiding striking with the different obstacles and also to minimize the time taken to cover from one part to another.

An introduction to automated motion planning for robots. First, a beginning path consists of a direct line connection in the configuration space from start to target is created. A novel point is inserted and pushed out of the collision at the most colliding configuration. For all of this, using only a matching skeletal robot and a point-like barrier, the design space is estimated. While using the reverse kinematics of a skeleton robot, a collision detection path for this estimation is created. This paper is divided into different sections as literature review, reactive approach, Genetic Algorithm, Conclusion

2. Literature Survey

D. Foxet al. [1] explored the built for robotic systems fitted with synchrony drives, which is actually derived from the movement mechanics of the robot. In studies, the external strategy securely operated the robot navigation RHINO at speeds up to 95 cm/sec, in an occupied and changing environment.

W. Sheng et al.[2] execute an area exploration mission, which recommends a convenient and accurate multi-robot coordination algorithm, provided that each robot's coverage area is small. To organize the activity of multiple robots, this method is applied to the data bidding model. To facilitate limited-range communications, two initiatives have been developed. First, in the bidding algorithm, the ranges between robots are taken into account, so that the robots appear to remain close to each other. Second, a system synchronization method is demonstrated to reduce the exchanged data volume when robot sub=-networks merge, centered on an innovative series amount map representation and efficient robot map update monitoring.

R. Rocha et al.[3] implemented a stochastic system for perception 3-D mapping, a distributed architecture in which each robot is dedicated to cooperating by information sharing with other robots. It determines an entropy-based measure of information utility that a robot uses to communicate the most valuable measurements to its teammates, thereby preventing the robot from overwhelming redundant information communication services.

A. Martinoliet al.[4] A time-discrete, gradual technique for modeling the mechanisms of distributed control experiments at the microscopic levels using hordes of humanoid systems equipped with reactive controllers. For semi metrics, the approach is well adapted because it does not take into consideration robot trajectories or the spatial distribution of objects. The power of the technique lies in the fact that it was created by considered progressive abstract measures, from actual robots to statistical models, each with well-defined mappings between gradual levels of implementation.

D. F. Wolf et al. [5]proposed an online algorithm for localizing and modelling complex environments simultaneously. Our technique is utilized for distinguishing and accurately representing static and dynamic parts of the world on the graph. Our approach is based on the maintenance of two grids for occupation. The stationary parts of the atmosphere are modelled with one grid, and the other versions are dynamic sections of the environment.

B. Yamauchi et al. [6] explore the boundary of the mobile robots between explored and unexplored environments. It also helps internal environmental conditions like walls, floors, etc.

D. Hahnel et al.[7] proposed the innovative algorithm that incorporates particle filtering and test fitting from Rao-Blackwellized. To reduce odometric errors during analysis, a match is often used in our method scan. For the resembling steps, a probability of the error terms of the test matching process is then used. The number of samples needed is severely reduced in this way. At the very same time, we were reducing the issue of particle loss that normally prohibits the robot from completing wide loops.

N. Siggelkowet al.[8]shows the Divide to conquer temporarily: Hierarchical, decentralized, and reintegrated organizational approaches to organizational research discovery and adjustment

J. Vazquez et al.[9]shows the companies which need to find operation configurations that are not always externally compatible, but also acceptable given the current climate of the market, in order to build a competitive advantage. After businesses have undergone an environmental shift that has changed the current business environment and established new, strong sets of activity options, this problem is especially acute.

M. N. Rookeret al.[10] show the discovery, which is a core robotics activity, a multi-robot system can be greatly advantageous. Domains of use provide monitoring, reconnaissance, space discovery, or rescue missions. The performance overall could be much quicker and more stable by using a group of robots. This paper presents an approach to multi-robot exploration that takes into account the limitations of wireless networking.

Challani, S. et al.[11] demonstrate the garbage system in the smart way of utilizing the waste resource of producing electricity and useful products.

N. Roy et al.[12] propose many alternate algorithms that robots might use when continuing to explore in attempting to rendezvous quickly. These algorithms exemplify various strategy groups whose relative suitability depends on the problem definition's characteristics. With regard to both expected- and worst-case behavior, we critically find the performances of the proposed algorithms. Also using numerical analysis and also a simulation of multi-agent exploration and rendezvous, we then analyze their behavior under a broader set of conditions.

A. Pal et al.[13] show a multi-robot exploration approach is presented for use in wireless environments. In order to have an accurate map of the environment for each moment and have an effective navigation plan to move into the unknown area, the challenges generally faced by a robot team are to retain network communication between themselves. We concentrate on incorporating these communication restrictions to resolve these problems and take into account navigation plan issues.

F. Cabrera-Mora et al.[14] show the multi-robot exploration framework seeks to reduce the discovery time and to decrease the robots' total travel distance by controlling the motion of the discovery robots. Modeling the world as a tree, we consider a model of coordination that reduces the number of robots enabled before each step to cross an edge and reach a vertex. This teamwork is accomplished by the robots using a series of active landmarks in a decentralized way that is dropped by them at the vertices explored.

S. M. LaValleet al.[15] show the motion path planning of a multi-robot system using the random tree approach.

B. Donald et al.[16] shows the motion path planning of different robot system using kindo-dynamics

M. Weiglet al.[17] show a mapping device fitted with an ultrasonic range finder is provided for the robot manipulator. A network of cells, which can be either free or filled, defines the world. It is suggested that data from the sonar be processed in two stages: first, the measurements are processed and assembled into the input layer, then both of these are aggregated into the world chart. The Shafer theory-based formulas are used in the model to correct for human error.

M. Likhachevet al.[18] show therandom time dynamics for the robot motion planning

R. C. Arkinet al.[19] show the multi-level description and scheduling capabilities are provided in the Autonomous Robot Architecture.The role of navigation systems path planning is discussed which provides the robot with a path assured to be free of collisions with any model obstacles. It is also possible to embed information supporting visual perception, enabling the actual route traversal by the vehicle.

Y. Shi et al.[20] show the complete study of particle swarm optimization in different fields of application mainly in robotics.

D. Anderson et al.[21] proposed theglobal change in the aspects of the Environments by the methods called Quaternary which allow the exploration in a change of environment.

M. F. Akayet al.[22] proposed that a medical disease like breast cancer will be supported by different vector machine Support vector machines along with different selection procedure

3. Social Significance

The major concern in the present scenario is to have smooth control of the robot's motion path. The avoidance of path obstacle on the way of robotics is the major serious issue which has to be rectified. Robotics has broad applications like military, medical, etc. In the existing methods shown in the literature review, the biggest problem with robot motion planning is smooth motion controlling of robotics without striking with obstacle and time minimization of the robot motion from one coordinate to another. The problem of obstacle avoidance is discussed with a reactive approach and time minimization with a genetic algorithm.

4. Reactive Approach for Robot Motion Planning

The scientist Britainberg formulate an approach that consists of a four-wheel system for navigating the path smoothly in an unknown static environment. This approach responds when the sensory data is interpreted and the wheel speed is changed. The technique for this work is to achieve the objective safely by avoiding stationary environmental barriers. During the test, various basic barriers were successfully tested in different static environments. For the computation of speed of wheels, mathematical descriptions is given as follows

The output of LIDAR sensor can be defined by

$$D_i = f(i) \qquad (1)$$

Where D_i is the distance of the obstacle in front of robot with i^{th} angle,

where i= 0,1,2- - - -- -- --- -- - - -180.

The obstacle value is a value that determines the net speed of the robot. If the obstacle value is high, the robot's net speed will be low and its opposite will also apply. To understand motion planning, first of all, we have to understand the following parameters associated with motion planning.

1) Obstacle value
2) Speed Factor

The **obstacle value (Ot)** gives the combined effect of density and distance of the obstacle in front of the robot and its angle. The closer the obstacles are to the robot and are in front, the higher its obstacle value. The more area in front of the robot with obstacles are present, in such a situation, the obstacle value will also be higher. The obstacle value is the sum of the right and left obstacle values.

The **speed factor (α)** is proportional to the speed of the robot, if the obstacles are more in front of the robot, then its obstacle value will be higher but its speed factor will be less.

The effect of sensors values on the obstacle value

$$Ot = \sum_{i=0}^{i=180} \omega_i \, d_i \qquad (2)$$

Where ω_i and d_i are the weight factor and sensor values respectively at an angle i. Generally, ω_i is used to determine the wheel speed and is given by

$$\omega_i = \frac{e^{\frac{(i-\mu)^2}{2\sigma}}}{\sigma\sqrt{2\pi}} \qquad (3)$$

Where

μ = mean value of angle, σ = standard deviation, i = angle (from 0 to 180), $\mu = 90^0$, $\sigma = 36^0$

Sensor value d_i is given by

$$d_i = (1 - \frac{D}{M}) \qquad (4)$$

Where Di is the distance of the obstacle from robot. For LMS291, maximum value of distance (M) is 80 meter as maximum range of sensor for LMS291 is 80 meters.
When any obstacle is at a distance less than 5 meter from the robot (Di< 5)
Obstacle value for each left and right wheel is given by Eq.(5) and Eq.(6) as

$$Ol = \sum_{i=0}^{i=90} \omega i \, di \quad (5)$$

and

$$Or = \sum_{i=90}^{i=180} \omega i \, di \quad (6)$$

Total Obstacle value (Ot) is given by Eq.(7) as

$$Ot = \sum_{i=0}^{i=90} \omega i \, di + \sum_{i=90}^{i=180} \omega i \, di$$

(7)

Now Ot helps in determining the value of speed factor α as given in equation (8)

$$\alpha = (1 - 0.9Ot)M_s / Ot \quad (8)$$

Where α is speed factor which generally lies in the range of 10 to 90 and Ms is maximum speed of robots (0.7 m/s).
Actual Left wheel Speed=speed factor * Obstacle value of left wheel

$$= \alpha \quad Ol \quad (9)$$

Actual Right wheel Speed=speed factor * Obstacle value of right wheel

$$= \alpha \, Or \quad (10)$$

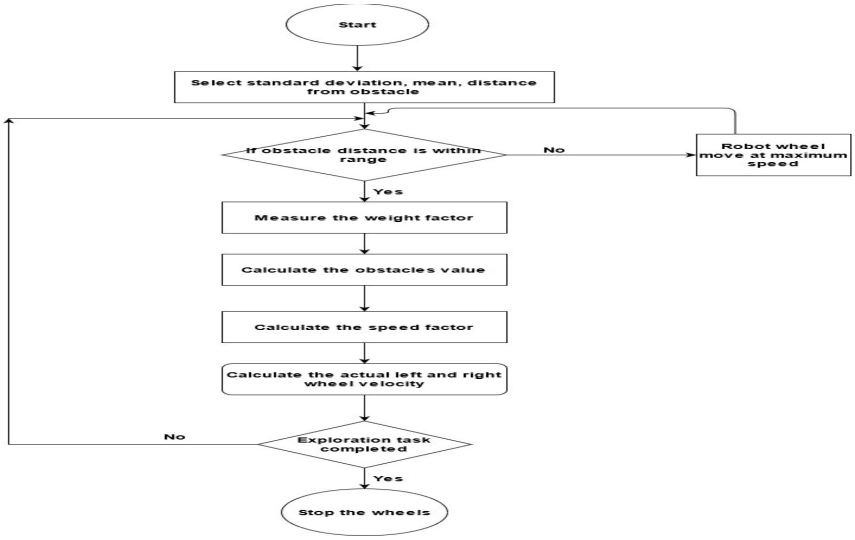


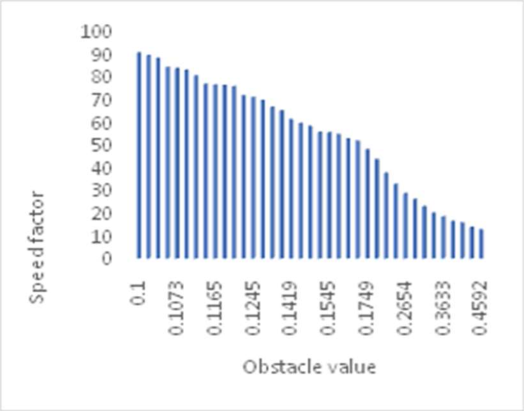
Fig. 1. Flowchart for calculation of speed

Equation (9) and (10) shows the actual wheel speed only in case of presence of obstacles at distance less than 5m, otherwise the robots are required to move with cruising speed (Ms=0.7 m/s) . Equation (7) clearly shows that if an obstacle is in front of robot or nearer to subordinate robot then Ot will increase and α will decrease it means speed of the robot wheel will decrease.
The whole process of computation of the speed and, is shown in Fig.1

4.1 Simulation with Webots

- **Variation of Wheel Velocity with angle of Obstacle between 0 to 90 Degree**:

The variation of speed factor with obstacle value is shown in Fig.2 and variation of left wheel speed with obstacle value is shown in Fig.3.



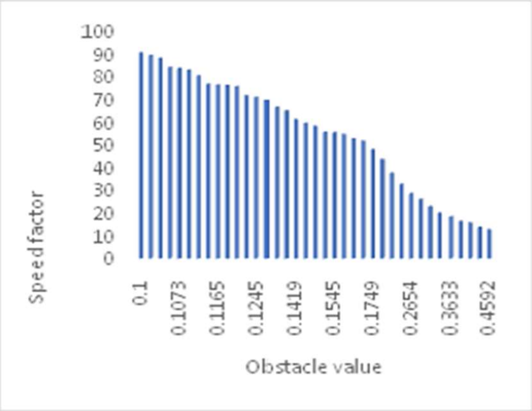
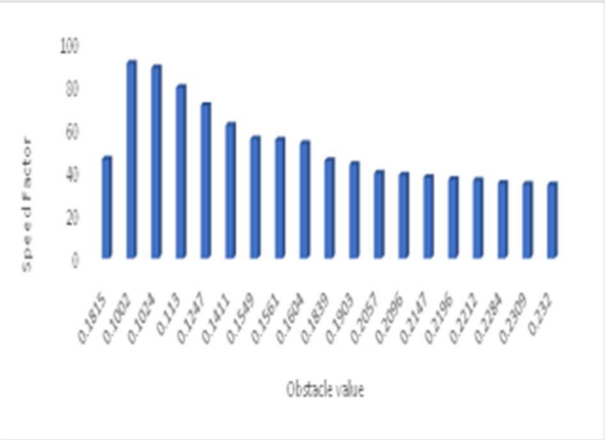


Fig. 2 Speed factor variation with obstacle value   Fig. 3. Left wheel speed with variation of obstacle value

- **Variation of Wheel Velocity with Angle of Obstacle from 90 to 180 Degree:**

The variation of speed factor with obstacle value, right wheel speed with obstacle value is shown in Figure 4. and Figure 5.



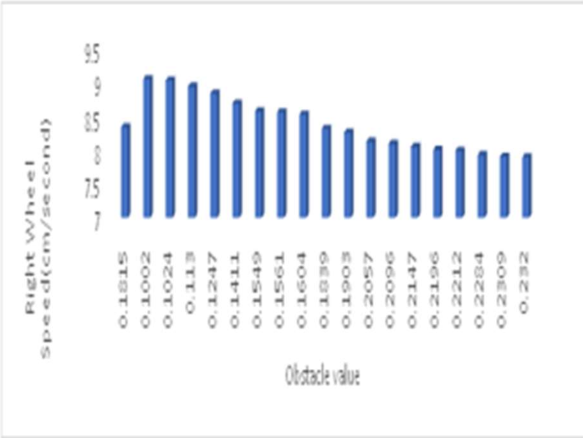


Fig. 4. Speed factor variation with obstacle value   Fig. 4. Right wheel speed with variation of obstacle value

- **Variation of Wheel Velocity with Angle of Obstacle from 0 to 180 Degree:**

In this case, the performance parameter of robot kinematics is analyzed when the obstacle is present on both sides. In this case, right obstacle and left obstacle with obstacle value in each side is shown in Fig.6. and while in Fig.7, variation of parameters like right wheel speed, left wheel speed, with obstacle value.

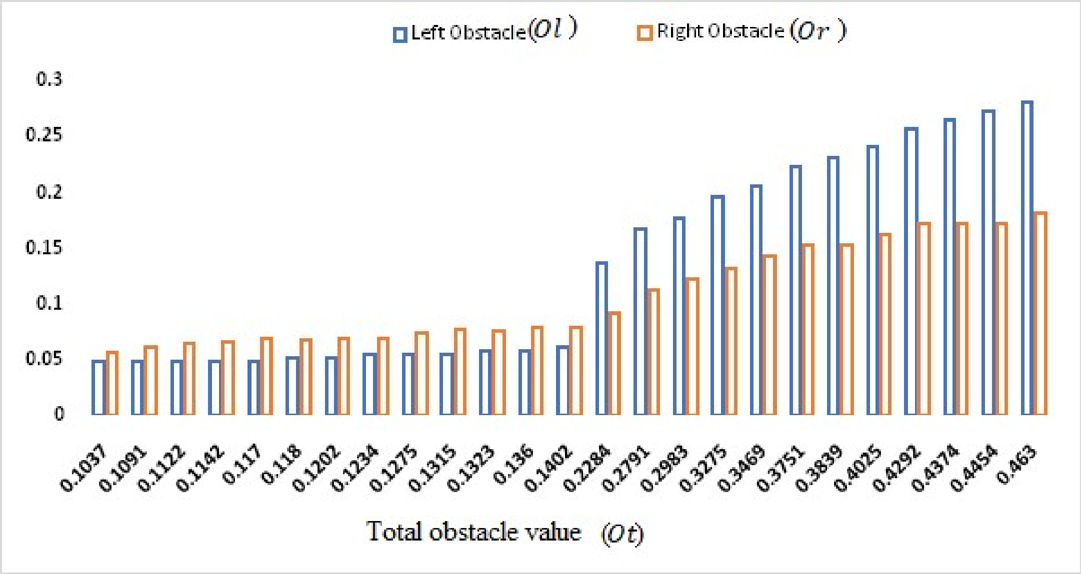


Fig. 6. Relation between total obstacle value and individual obstacle value

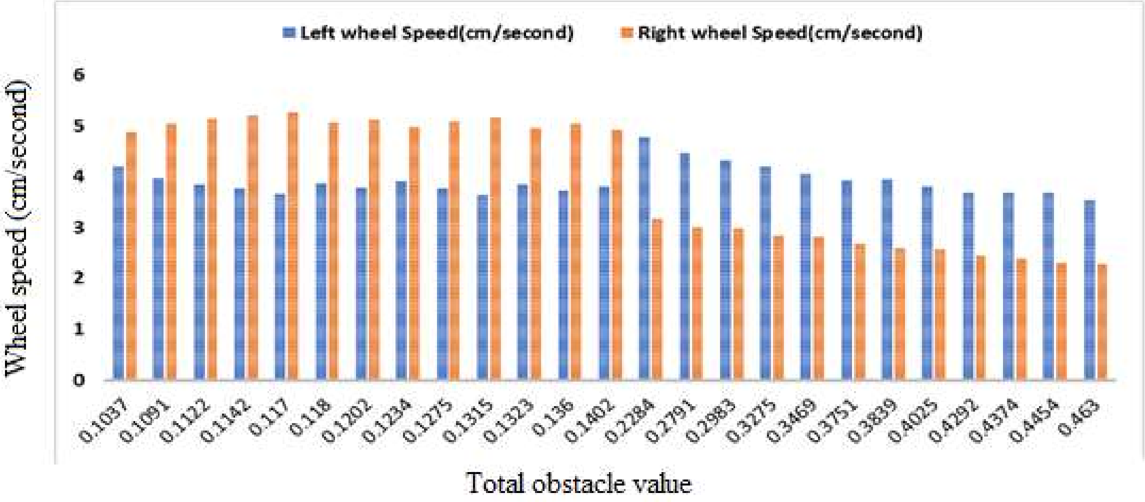


Fig. 7. Variation of wheel speed with total obstacle value

It is observed that when the obstacle is at the centre the speed of the right side wheel drastically decreases from its peak value while the speed of the left side wheel drastically increases to peak value so that striking of an object with an obstacle can be avoided.

From Fig.6. and Fig.7. the variation of wheel speed is observed with the different obstacle value. Looking at these figures, it is clear that the direction in which the value of the obstacle increases, the speed of the wheel of that direction increases.

5. Genetics Algorithm for Time Minimization of Robotics Path

Till now the effect of the obstacle on robot motion is discussed. In this section time taken by the robot to travel from one place to another will be analyzed. In order to consider the time effect, a Genetic Algorithm is used.

At the University of Michigan, USA, John Holland developed a genetic algorithm in the 1970s, allowing computers to create solutions to difficult search problems and combinatorial problems. The genetic algorithm follows the behaviour of natural genetics. It consists of three processes: reproduction, crossover, mutation.

The robot is at position(x, y) with initial coordinates (0, 0) at an angle (θ) but initial coordinates can also vary. Even the robot can walk in three dimensional systems as well. The general coordinator of the robot is (x, y, θ).

Let the initial position is $x_{t-1}$and after a certain time (t) robot occupy position x_t with velocity v_t

Sometimes, velocity is also called motion command .The conditional probability density for finding the location of the robot is given as P $(x_t/v_t, x_{t-1})$.The velocity motion model of the robotics kinematics is represented as (v_t, ω_t) where v_t is positive translational velocity, ω_t is positive rotational velocity.

For the analysis of robot kinematics, the layout of the robot structure with four coordinates is shown in Fig.8.

It is clear from Fig.11 that initially robot is at coordinates (x, y, θ).) & after a certain time‘t’ robot takes new coordinates(x, y, θ).Two positions of the robot at a certain angle are measured with respect to the x-axis. Take arbitrary position (x*,y*) which shows the random location of the robot which we have to find out. The Certain calculation is required for measuring the arbitrary position (x*,y*) refer to Fig.8. is required which is shown from Eq.11 to Eq.15

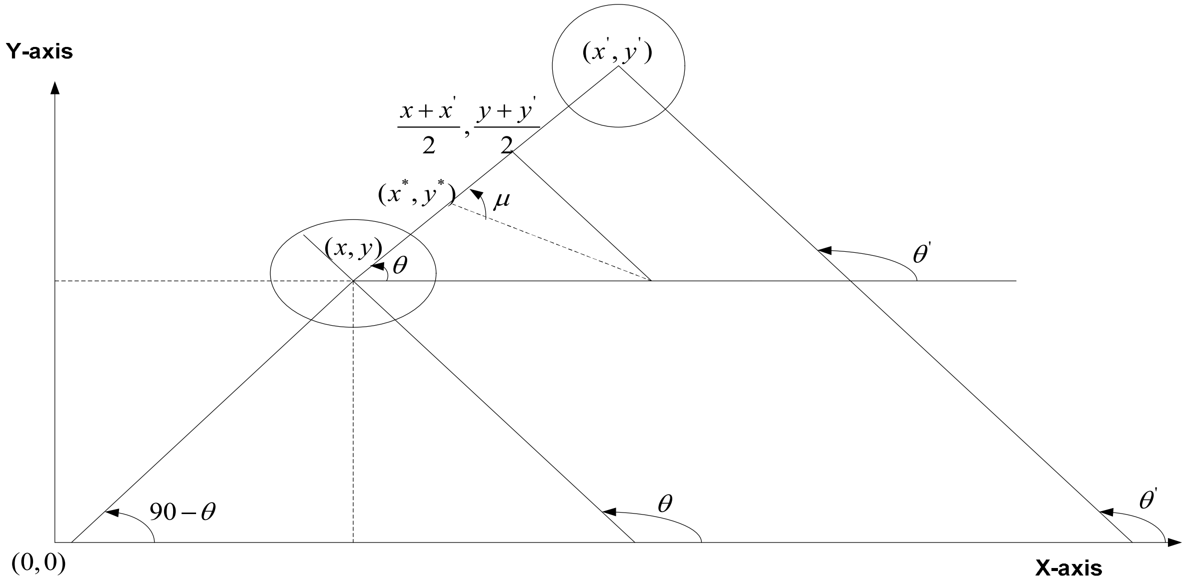


Fig.11Working Model of the Robot

$$\mu = \frac{1}{2}\frac{(x-x')\cos\theta+(y-y')\sin\theta}{(y-y')\cos\theta-(x-x')\sin\theta} \tag{11}$$

$$x^* = \frac{x+x'}{2}+\mu(y-y') \tag{12}$$

$$y^* = \frac{y+y'}{2}+\mu(x'-x) \tag{13}$$

$$r^* = \sqrt{(x-x^*)^2+(y-y^*)^2} \tag{14}$$

$$\Delta\theta = \tan^{-1}(\frac{y'-y^*}{x'-x^*})-\tan^{-1}(\frac{y-y^*}{x-x^*}) \tag{15}$$

It is known that the robot moves both translatory as well as rotatory. Relation between linear & rotational speed is given in Eq.16 & Eq.17 as

$$v = r^*\omega \tag{16}$$

$$\omega = \frac{\Delta\theta}{\Delta t} \tag{17}$$

Put the value of ω in Eq.16then Eq.17 is attained

$$v = r^*\frac{\Delta\theta}{\Delta t} \tag{18}$$

After arranging the above term Eq.19 will be attained

$$\Delta t = r^*\frac{\Delta\theta}{v_t} \tag{19}$$

Robot motion begins at t=0, then change in time is given as $\Delta t = t-0 = t$ & can also be written as

$$t = r^*\frac{\Delta\theta}{v_t} \tag{20}$$

If we put all the values from Eq.11 to Eq.15 in Eq.20, then it can be realized that time travel t depends on five parameters initial coordinates, final coordinates, angle(x, y, x , y’, θ).
But the representation of initial & final coordinates get changes in every new path while for the angle it will remain the same but its value may vary.
It can be pointed to from fig.8. that the robot can travel in a specific area and change initial coordinates (x', y') & final coordinates (x^*, y^*) every time and replace it with previous initial & final coordinates in the calculation section from Eq.11 to Eq.20. Let the coordinates be taken at an angle between

$(x,y)=(2,3)$, $(x',y')=(3,6)$, $(x^*,y^*)=(6,7)$, $(x'',y'')=(5,5)$

at an angled lie between $0 < \theta < 60^{\circ}$

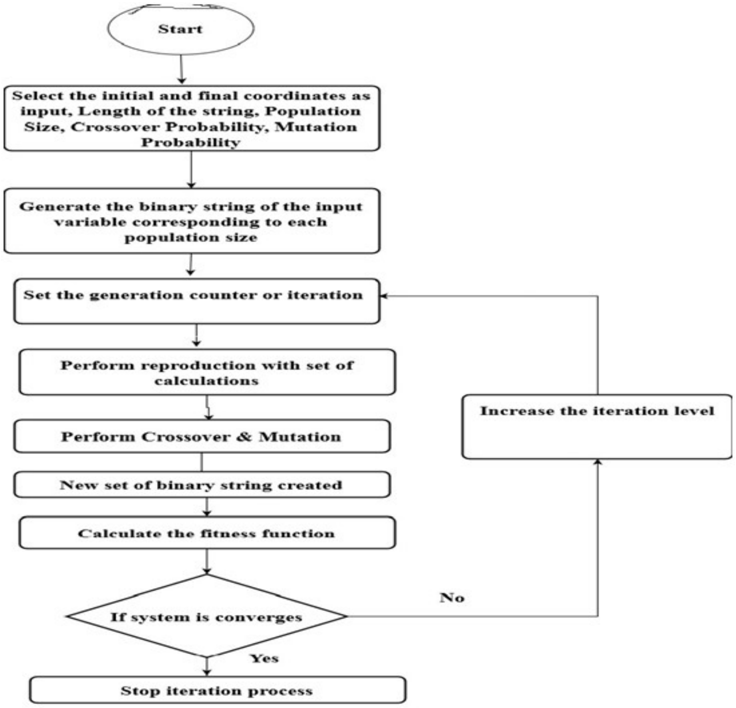


Fig.9. Flowchart Representing the GA Procedure for Robot Kinematics

The distance has been taken in meters and speed of robot has been assumed as 10 m/s. The process of analyzing the motion control from one place to another will be assessed by using a genetic algorithm is explained through the flow chart in Fig.9

Let's analyze the robot motion from initial coordinates (x, y) = (2, 3) to (x', y') = (4, 6) by using the flowchart.
Let decide the basic parameter for the reproduction operator
Population size: 5
Length of each variable: 3
Length of the complete string: 15
Crossover Probability, P_c=0.8, Mutation Probability, P_m=0.02
By considering the Eq. (20) as the fitness function for time evaluation to move from one coordinate to other

Table.1Time Taken by Robot for First Iteration Table.2 Time Taken by Robot after 50 Iteration

Population	Time(sec)
1	0.0370
2	0.0410

3	0.0353
4	0.0385
5	0.0305

Population	Time(sec)
1	0.0310
2	0.0369
3	0.0301
4	0.0312
5	0.0380

Table.3 Time Taken by Robot after 133 Iteration

Population	Time(sec)
1	0.0370
2	0.0371
3	0.0371
4	0.0371
5	0.0371

It has been observed from Table 1 to Table 3 that the system converges as the iteration frequency increases. It was noted that after 133 iterations, the time is taken by the robot to move from (2,3) to (3,6) eventually converges. The time taken to move by the robot is 0.0372 seconds. The various characteristics of GAs are shown in Fig.10.

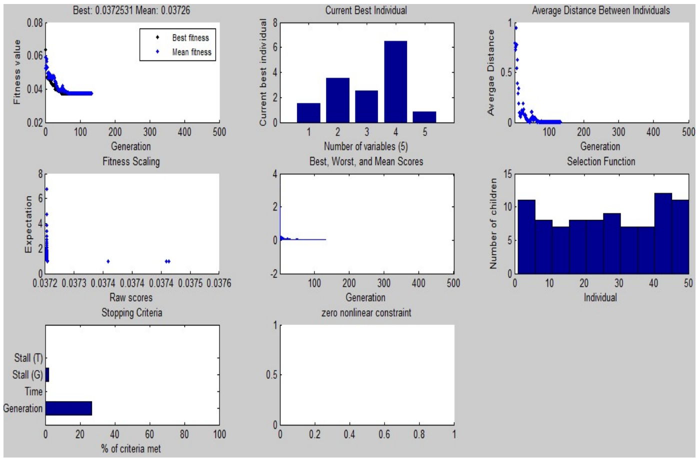


Fig.10 Different Characteristic of GAs for Distance from (2,3) to (3,6)

Following the full GAs operator protocol above. The time taken by the robot to move from (3,6) to (6,7) can also be found in Fig.11. As shown in Table.4, it is found that the robot takes 239 iterations to complete the journey from (3,6) to (6,7)

Table.4Time Taken by Robot after 239 Iteration

Population	Time(sec)
1	0.127
2	0.128
3	0.127

4	0.127
5	0.127

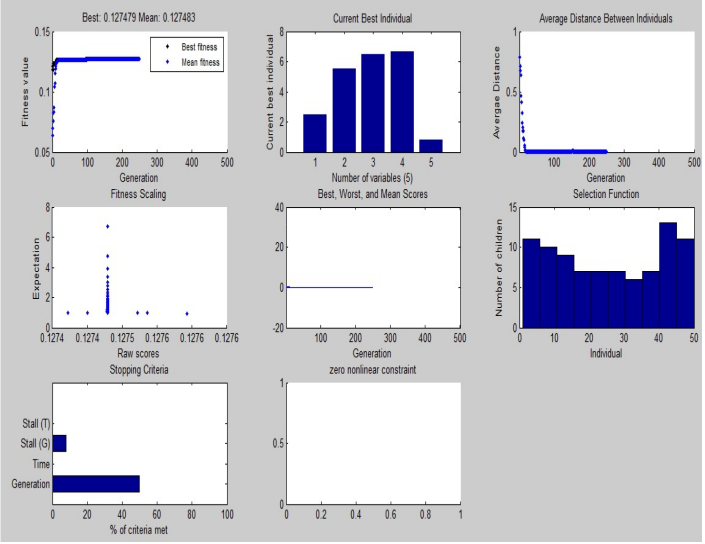


Fig.11 shows the GAs for distance from (3,6) to (6,7).

It is also possible to find the time taken by the robot to travel from (6,7) to (5,5) from Fig.15. It is found that the robot is taking 135 iterations to complete the journey from (6,7) to (5,5) as shown in Table.5

Table.5Time Taken by Robot after 135 Iteration

Population	**Time(sec)**
1	0.32
2	0.33
3	0.33
4	0.33
5	0.33

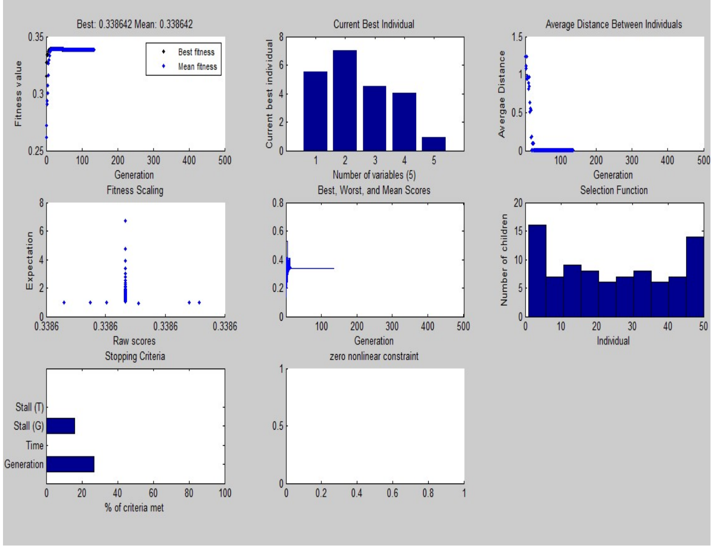


Fig.12 Different Characteristic of GAs for Distance from (6,7) to (5,5)

Fig.12 shows the GAs for distance from (6,7) to (5,5). It is also possible to find the time taken by the robot to travel from (5,5) to (2,3) from Fig.13. It is found that the robot is taking 62 iterations to complete the journey from (5,5) to (2,3) as shown in Table.6

Table.6 Time Taken after 62 Iteration

Population	**Time(sec)**
1	0.342
2	0.342
3	0.341
4	0.342
5	0.342

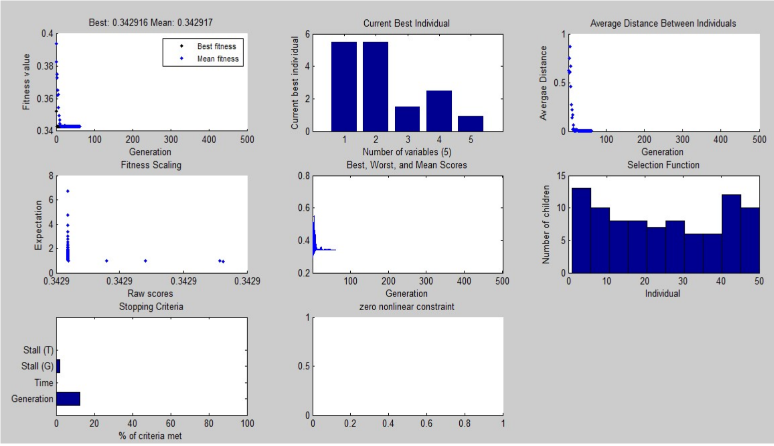


Fig.13.Different Characteristic of Robot Kinematics for GAs of Distance from (5,5) to (2,3)

Now we compare the performance of robot kinematics with GAs and existing methods as discussed in Table.7

Table.7 Comparative analysis between with GAs & existing methods for the time taken by the robot to travel in coordinates

Travel Coordinates(meter)	Time taken (sec) by Robot with GAs	Ref.[3]	Ref.[9]	Ref.[15]
$(x\ ,y\ )$ to $(x^{'},y^{'})$	0.0371	0.051	0.062	0.071
$(x^{'},y^{'})$ to (x^{*},y^{*})	0.127	0.215	0.239	0.312
(x^{*},y^{*}) to $(x^{''},y^{''})$	0.33	0.285	0.365	0.412
$(x^{''},y^{''})$ to $(x\ ,y\ )$	0.342	0.366	0.355	0.388

It is observed from Table.7 that time taken by robots to travel with GAs is lesser with existing techniques (Ref.[3], Ref.[9],Ref.[15]). This shows the novelty of the proposed scheme Genetic Algorithm over the existing technique.

6. Conclusions

The paper shows the analysis of robot motion controlling optimally through a reactive approach and genetic algorithm. The avoidance of striking with hurdles is the major challenge of robotics due to which robot deviates from its path and reached the wrong destination. This problem is rectified by using the reactive approach method. In this method, a proper system is designed in which the left wheel and right wheel speed are controlled and prevent the striking of robots from obstacle smoothly so that robot reached its desired destination safely. In addition to this, the time taken by robots to move from one place to another is being decided by a genetic algorithm. The utility of genetic algorithms in the design model of robotics helps in minimizing the time taken to travel with the existing methods.

References

[1] D. Fox, W. Burgard, S. Thrun et al.(1997). The dynamic window approach to collision avoidance," IEEE Robotics & Automation Magazine, vol. 4, no. 1, pp. 23–33.

[2] W. Sheng, Q. Yang, J. Tan, and N. Xi(2006) Distributed multi-robot coordination in area exploration," Robotics and Autonomous Systems, vol. 54, no. 12, pp. 945–955.

[3] R. Rocha, J. Dias, and A. Carvalho(2005). Cooperative multi-robot systems:: A study of vision-based 3-d mapping using information theory," Robotics and Autonomous Systems, vol. 53, no. 3, pp. 282–311

[4] A. Martinoli, K. Easton, and W. Agassounon (2004) Modeling swarm robotic systems: A case study in collaborative distributed manipulation," The International Journal of Robotics Research, vol. 23, no. 4-5, pp. 415–436.

[5]D. F. Wolf and G. S. Sukhatme(2005). Mobile robot simultaneous localization and mapping in dynamic environments," Autonomous Robots, vol. 19, no. 1, pp. 53–65.

[**6**] B. Yamauchi(1998) Frontier-based exploration using multiple robots," in Proceedings of the second international conference on Autonomous agents. ACM, pp. 47–53.

[7] D. Hahnel, W. Burgard, D. Fox, and S. Thrun(2003) An efficient fastslam algorithm for generating maps of large-scale cyclic environments from raw laser range measurements," in Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on, vol. 1. IEEE, pp. 206–211.

[8] N. Siggelkow and D. A. Levinthal(2003) temporarily divide to conquer: Centralized, decentralized, and reintegrated organizational approaches to exploration and adaptationOrganization Science, vol. 14, no. 6, pp. 650–669.

[9] J. Vazquez and C. Malcolm, "Distributed multirobot exploration maintaining a mobile network," in Intelligent Systems, 2004. Proceedings. 2004 2nd International IEEE Conference, vol. 3. IEEE, 2004, pp. 113–118.

[10] M. N. Rooker and A. Birk (2007) Multi-robot exploration under the constraints of wireless networking," Control Engineering Practice, vol. 15, no. 4, pp. 435–445.

[**11**] Challani, S., Burde, D., Kutemate, H., Lanjewar, S., &Dorve, J. D. (2020). Smart Garbage Management System. *Recent Trends in Analog Design and Digital Devices*, *3*(1).

[12] N. Roy and G. Dudek(2001) Collaborative robot exploration and rendezvous: Algorithms, performance bounds and observations," Autonomous Robots, vol. 11, no. 2, pp. 117–136.

[13] A. Pal, R. Tiwari, and A. Shukla (2012) Multi-robot exploration in wireless environments, Cognitive Computation, vol. 4, no. 4, pp. 526–542.

[14] F. Cabrera-Mora and J. Xiao(2012) A flooding algorithm for multirobot exploration, Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, vol. 42, no. 3, pp. 850–863.

[15] S. M. LaValle (1998) Rapidly-exploring random trees a new tool for path planning.

[16] B. Donald, P. Xavier, J. Canny, and J. Reif, "Kinodynamic motion planning(1993) Journal of the ACM (JACM), vol. 40, no. 5, pp. 1048–1066.

[17] M. Weigl, B. Siemiatkowska, K. A. Sikorski, and A. Borkowski, "Grid-based mapping for autonomous mobile robot," Robotics and Autonomous Systems, vol. 11, no. 1, pp. 13–21, 1993.

[18] M. Likhachev, D. I. Ferguson, G. J. Gordon, A. Stentz, and S. Thrun(2005) Anytime dynamic a*: An anytime, replanning algorithm." in ICAPS, pp. 262–271.

[19] R. C. Arkin(1999) Navigational path planning for a vision-based mobile robot," Robotica, vol. 7, no. 01, pp. 49–63.

[20] Y. Shi and R. C. Eberhart(1999) Empirical study of particle swarm optimization," in Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on, vol. 3. IEEE.

[21] D. Anderson, A. Goudie, and A. Parker(2013) Global Environments through the Quaternary: Exploring Environmental Change. Oxford University Press.

[22] M. F. Akay(2009) Support vector machines combined with feature selection for breast cancer diagnosis," Expert systems with applications, vol. 36, no. 2, pp. 3240–3247, 2009

[23] Danthala, S. W. E. T. H. A., Et Al. "Robotic Manipulator Control By Using Machine Learning Algorithms: A Review." *International Journal Of Mechanical And Production Engineering Research And Development* 8.5 (2018): 305-310.

[24] Mengistu, Abrham Debasu, And Dagnachew Melesew Alemayehu. "Robot For Visual Object Tracking Based On Artificial Neural Network." *International Journal Of Robotics Research And Development (Ijrrd)* 6.1 (2016): 1-6.

[25] Saravanan, S., S. S. Aravinth, And M. Rameshkumar. "A Novel Approach In Agriculture Automation For Sugarcane Farming By Human Assisting Care Robot." *2017). International Journal Of Agricultural Science And Research (Ijasr) 7.4, Aug 2017, 107* 112 (2017).

[26] Kumar, S. Ramesh, Et Al. "Design And Fabrication Of Autonomous Robot For Precision Agriculture." *Int. J. Mech. Product. Eng. Res. Develop* 8.3 (2018): 385-392.

[27] Deshmukh, Dinesh1 B., And Praveen Kumar Bhojane. "Design Of Robotics Gripper Using Cad Software And Manufacturing For Tal Brabo Robot." *International Journal Of Mechanical And Production Engineering Research And Development (Ijmperd) 10, Special Issue, Aug 2020, 100* 105.

[28] Jose, J. A. I. S. E., Et Al. "A Survey On Magnetic Wall-Climbing Robots For Inspection." *Transst. J* 8 (2018): 59-68.