Dynamic Point Bug Algorithm For Robot Navigation

B. Margaret Devi, Prabakar S

ABSTRACT- The main task of a robot is to search a collision free path in order to reach the target specified. The main problem in robot navigation is localization i.e. the robot should know its present location. Here in this algorithm the localization problem is solved by using graphical method. This algorithm can be included in Bug Algorithm family. Dynamic point bug algorithm has been implemented for robot navigation system in NI-sbRIO 9631. This proposed work provides solution to identify the present location of the robot while moving towards target based on coordinates estimation.

Keywords- Coordinate estimation, Dynamic point bug algorithm, Localization, Navigation, Path Planning, sbRIO 9631, Target.

1. INTRODUCTION

Autonomous robots which work without human operators are required in robotic fields. In order to achieve its goal, autonomous robots have to be intelligent and should decide their own action. When an autonomous robot moves from a point to a target point in its given environment, it is necessary to plan an optimal and feasible path avoiding obstacles in its way. So, path planning is one of the important factors to be considered while developing an autonomous robot. Path planning gives the path to be followed by the robot in order to reach the target. The dynamic point bug algorithm introduced in this paper helps in determining the obstacle free path that the robot has to travel, this algorithm uses geometry and mathematical formula to get the desired path.

- *B Margaret Devi is currently pursuing master's degree program in Embedded System in Karunya University, India, PH-9486679252. E-mail: margaretsharma@gmail.com*
- *Prabakar S is currently working as an Assistant Professor in Electronics and Instrumentation Engineering Department in Karunya University, India, PH-9150160947. E-mail: srisornaprabu@gmail.com*

2. DYNAMIC POINT BUG ALGORITHM

Dynamic point bug algorithm can be included in the bug algorithm family. The dynamic point bug algorithm allows the robot to navigate in the given environment and it will avoid obstacle while it travels towards the target. The robot is equipped with sonar sensor i.e. an ultrasonic sensor which helps in detecting an obstacle. The robot will move according to the straight line (L) joining the current position and the target position.

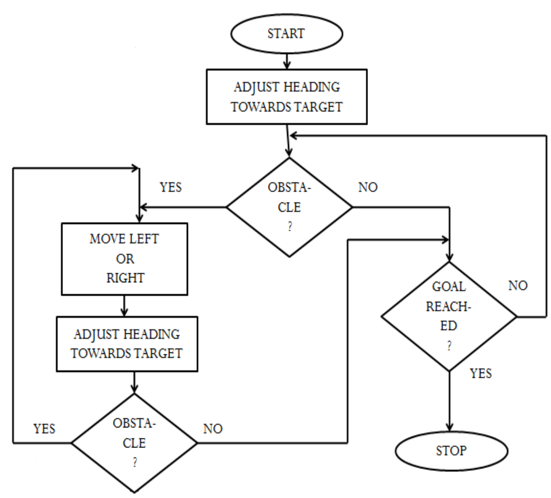

Fig1. Dynamic point bug algorithm

In the beginning it will move towards the target, when an obstacle is detected the robot will avoid the obstacle and a new position will be obtained now the robot will calculate the angle between the current position and the target position and it will start moving with the new angle again if an obstacle is has to travel so as to reach the destination.

3. DYNAMIC POINT BUG ALGORITHM ANALYSIS

The main goal of this algorithm is to generate a collision free path so that the robot will reach the target. The robot will travel from source(S) to destination (D) which is fixed.

As the source point and the destination point is fixed we can calculate the shortest distance between this two points which is a straight line joining this two points. So, we can calculate the distance from the line equation,

$$L = \sqrt{(x_t - x_c)^2 + (y_t - y_c)^2} \quad (1)$$

And we can also calculate the angle at which the target is located from the source point.

$$\text{Slope (m)} = \frac{y_t - y_c}{x_t - x_c} \quad (2)$$

$$\theta = \tan^{-1} m \quad (3)$$

As the robot travel towards the target, if an obstacle is detected it will avoid it and a new position will be estimated and the angle between the new position and the target will be calculated and the robot will now move with that angle. It will repeat this until it reaches the target. The robot will continuously compare the current (x_c,y_c) position and the target(x_t,y_t), When this two values will become equal the robot will identify that point as the target point and will stop.

Also the robot is continuously calculating the shortest line between the points when the shortest line value will be zero the robot will identify that point as the target point.

There will be two cases which can be stated as,

Case I- When the robot is in the origin

If the robot is in the starting position it will move with an angle which is given by,

$$\theta = \tan^{-1}\frac{y_t}{x_t} \quad (4)$$

Case II- When obstacle is detected

The robot will avoid the obstacle and move with some angle which is given by,

$$\theta = \tan^{-1}\{\frac{(y_t - y_c)}{(x_t - x_c)} \quad (5)$$

Where, (x_t ,y_t) is the target position and (x_c,y_c) is the current position of the robot.

4. SYSTEM REQUIREMENT

4.1 Hardware requirement

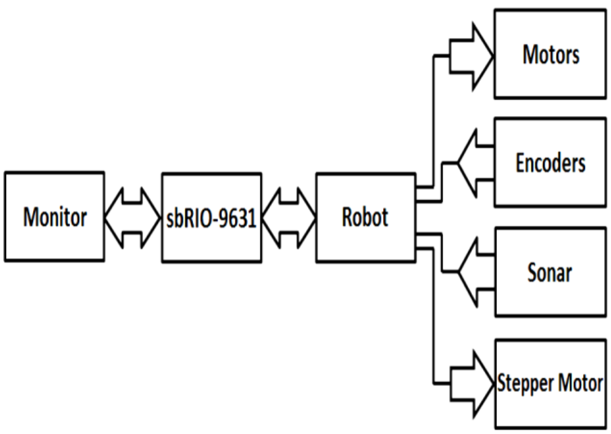

Fig: 2. Block diagram of sbRIO-9631

The NI Robotics Starter Kit is a mobile robot platform that comes equipped with sensors (sonar), motors, and an NI Single-Board RIO for embedded control. Fig 2 shows the NI Starter kit 9631. NI Lab VIEW graphical programming and the Lab VIEW Robotics module can be used for programming the mobile robot.

4.2 Software Requirement

LabVIEW (short for Laboratory Virtual Instrumentation Engineering Workbench) is a system design platform and development environment for a visual programming language from National Instruments. LabVIEW programs or subroutines are called virtual instruments (VIs). A VI has two components: a block diagram and a front panel. The Controls and indicators on the front panel allow an operator to input data into or extract data from a running virtual instrument. The front panel also serves as a programmatic interface. So, a virtual instrument can either be run as a program, with the front panel providing a user interface, or, when put as a node onto the block diagram, the front panel gives the inputs and outputs for the given node through the connector panel. This means each VI can be easily tested before being embedded as a subroutine into a larger program.

5. SIMULATION AND RESULTS

The simulation of dynamic point bug algorithm is carried out in LabVIEW robotics 2010.The algorithm will avoid obstacle, the position of the robot will be estimated and the angle at which the robot has to travel will be calculated using equation (2) and (3) in the meantime the shortest distance between the current position and the target position is also calculated when the robot reaches target this value will be zero. Distance L is calculated using equation (1) and the value of this L will be zero when it reaches target.

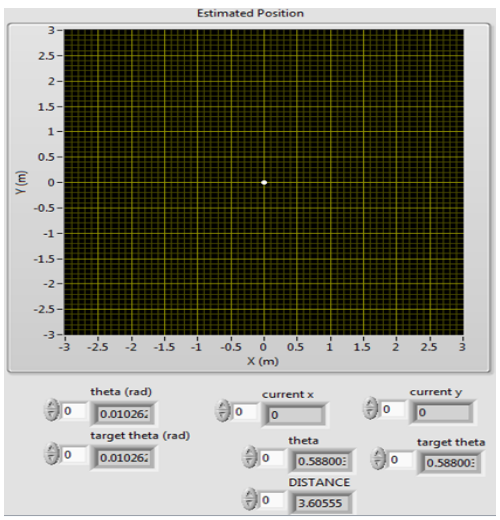

Fig:3 sbRIO position at the origin

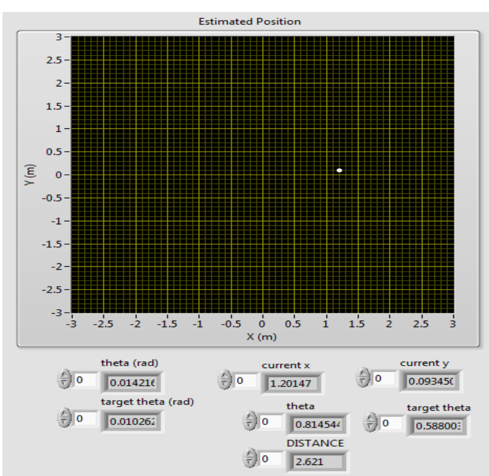

Fig: 4 sbRIO 9631 position after navigation

6. DISCUSSION AND CONCLUSION

A general roaming and obstacle avoidance algorithm has been developed with LabVIEW environment and embedded in the sbRIO9631 robot and has been evaluated that the robot could be navigate without colliding with any objects in the environment. A position estimation algorithm has also been accompanied with the roaming and obstacle avoidance program to estimate the current position of the robot in an unknown environment. This position estimation has been developed based on the graphical localization method which provides better position estimation of the mobile robot in an unknown environment. The angle with which the robot has to travel is calculated and travel with that angle so that it will reach the target position without collision.

7. ACKNOWLEDGEMENT

The authors would like to thank God Almighty for His guidance and presence throughout the work of this paper and also Dona Varghese and Gayathri S for their valuable and effective suggestions.

8. REFERENCES

[1] Woo Young Kwon, Il Hong Suh, and Sanghoon Lee, **"SSPQL: Stochastic Shortest Path-based Q-learning"**, International Journal of Control, Automation, and Systems,Vol.9, Issue 2,Pg No.328-338, 2011.

[2] Mohammad Abdel Kareem Jaradat, Mohammad Al- Rousan ,Lara Quadan, **"Reinforcement based Mobile Robot Navigation in Dynamic Environment"** , Science direct Journal of Robotics and Computer Integrated Manufacturing,Vol.27,Issue1,Page no. 135-149,2011.

[3] Buniyamin N., Wan Ngah W.A.J., Sariff N., Mohammad Z., **"A Simple Local Path Planning Algorithm for Autonomous Mobile Robots"**, International Journal Of Systems Applications, Engineering & Development,Volume 5, Issue2,Page no.151-159, 2011

[4] Yu- Cheol Lee, Wonpil Yu, Jong-Hwan Lim, Wan-Kyun Chung and Dong-Woo Cho, **"Sonar Grid Map Based Localization for Autonomous Mobile Robots"**, Mechatronics and Embedded systems and Application , IEEE International Conference ,Page no. 558-563, Oct. 2008

[5] O.Hachour, **"Path planning of Autonomous Mobile robot"**, International journal of systems applications, engineering & development, Volume 2, Issue 4,Page no.178-190, 2008

[6] Lauro Ojeda, Johann Borenstein, **"Personal Dead-reckoning System for GPS-denied Environments"**, IEEE International Workshop on Safety, Security, and Rescue Robotics, Rome, Italy, September 27-29, Page No. 1-6, 2007.

[7] Meng Joo Er, Member, IEEE, and Chang Deng, **"Obstacle Avoidance of a Mobile Robot Using Hybrid Learning**

Approach",IEEE transactions on industrial electronics, vol. 52, No. 3,Page No. 898-905, June 2005.

[8] Javier Minguez, Luis Montano, **"Nearness Diagram (ND) Navigation: Collision Avoidance in troublesome Scenarios"**, IEEE Transactions on robotics and automation, Vol.20, No.1,Page no.45-59,Feb. 2004

[9] Torvald Ersson and Xiaoming Hu, **"Path Planning and Navigation Of Mobile Robots in Unknown Environments"**, International conference on intelligent robots and systems, Vol.2, Page no.858-864, 2001

[10] Kai-Tai Song and Charles C. Chang, **"Reactive Navigation in Dynamic Environment Using a Multisensor Predictor"**, IEEE Transactions on Systems, Man, and Cybernetics—Part b: Cybernetics, Vol. 29, No.6, Page No. 870-880, December 1999

[11] Aminesh Chakravarthy and Debasish Ghose, **"Obstacle Avoidance in a Dynamic Environment: A Collision Cone Approach"**, IEEE Transactions On Systems, man, And Cybernetics—part A: Systems And Humans, Vol. 28, No. 5, Page No. 562-574,September 1998.

[12] J. Borenstein, H.R. Everett, L. Feng, and D. Wehe, **"Mobile Robot Positioning & Sensors and Techniques"**, Journal of Robotic Systems, Special Issue on Mobile Robots. Vol. 14, No. 4, 1997, Page No. 231 – 249, December 1999.

[13] Taner Bilgic¸ and I. Burhan Turksen, **"Model-based Localization for an Autonomous Mobile Robot Equipped with Sonar Sensor**s**",** System, Man and Cybernetics, IEEE Conference, Vol.4, Page no.3718-3723, Oct. 1995.

[14] J. Borenstein, Member, IEEE and Y. Koren, Senior Member, IEEE, **"The Vector Field Histogram -Fast Obstacle Avoidance For Mobile Robots",** IEEE Journal of Robotics and Automation, Vol 7, Issue 3, Page No.. 278-288, June 1991.

[15] Borenstein, Yorem Koren, **"Real-Time Obstacle Avoidance for Fast Mobile Robots"**, IEEE Transactions On Systems, Man, And Cybernetics, vol. 19, no. 5, Page No.1179-1187, September/October,1989.