



AN ALGORITHMIC APPROACH FOR OBSTACLE AVOIDING MOBILE AUTOBOT

Er. Shivanshu Shrivastava *¹

*¹ B.Tech, Electrical& Electronics Engineering, VIT University, Vellore, India.

Keywords: Obstacle avoiding, wheeled robot.

Abstract

An obstacle avoiding mobile Autobot, is a small wheeled robot that travels from a starting point to a destination point by avoiding collision with the various objects in its path. This paper proposes an algorithm for obstacle avoidance Robot based on the odometry that is the path or track of travel. It also formulates the implementation strategy in avoidance of obstacles, relative to the position of the robot. A two dimensional co-ordinate system is generated using odometry which helps the Autobot to remain over its desired path.

Introduction

The application of robotics field is broadly used in the field of research, laboratory based work, industrial work to automate process and reduce the human errors. This paper is describing the design of an Autobot. This paper proposes an algorithm for obstacle avoidance Robot based on the odometry that is the path or track of travel. It also formulates the implementation strategy in avoidance of obstacles, relative to the position of the robot. A two dimensional co-ordinate system is generated using odometry which helps the Autobot to remain over its desired path.

A number of techniques to create unmanned vehicles or robots have been proposed, which itself states the need of efficient navigation algorithms for such systems to work upon. This paper is an attempt in this direction.

Background

The concept of odometry has been used to determine a virtual co-ordinate system to determine the relative position of the autobot with respect to its previous position. [1] Mentions the various processes that can be used for odometry. Also, [2] explains the concept of odometry which can be used to determine the relative position of the autobot with respect to its previous position.

Construction

The construction includes a small circular disc mouthed over two electric motors used as wheels placed diametrically opposite and can be individually controlled by a micro-controller [3]. Two infrared transmitters and receivers are placed over each of the wheel positions. Another proximity sensor can be placed at the front end of the vehicle along the line parallel to the wheels and passing through the center of the circular disc. The wheels of the robot have wheel encoders which provide the necessary information about the position of the robot.

Understanding the Concept



Global Journal of Engineering Science and Research Management

It is of prime importance to know the position of the Autobot at any point of time. For that, we need to refer to odometry to find the position of an object. Suppose, the START position is considered to be as 'x0' for X-Coordinate and 'y0' for Y-Coordinate. If the object moves to a new point in a straight line at an angle 'α' then the new coordinates would be –

$$x1 = x0 + D \cdot \cos\alpha$$

$$y1 = y0 + D \cdot \sin\alpha$$

Hence, the coordinates of the wheels of the autobot can be determined as –

$$Xc = xc0 + Dc \cdot \cos\alpha$$

$$Yc1 = yc0 + Dc \cdot \sin\alpha$$

$$\alpha' = \alpha + (Dr - Dl) / L$$

Dc can be calculated as the mean distance travelled by the Left and Right wheel. If Dl is distance travelled by Left wheel and Dr is distance travelled by right wheel then Distance travelled by center of Autobot is

$$Dc = (Dr + Dl) / 2$$

Hence, position of the center of the autobot can always be determined.

Track to follow and obstacle avoiding

The track which the Autobot needs to follow should be pre-fed into the built in microcontroller, i.e., the distance and the angle of the final point from the initial point. If the Autobot needs to travel in a curved path the required equation of the curve needs to be provided. Once the equation is determined, the instantaneous coordinates of the center of the Autobot are fed into the equation over which the Autobot has to travel. The micro-controller calculates the position of the center of the Autobot on the basis of the information provided by the left and right wheel encoders and feeds these instantaneous co-ordinates to the desired equation of travel. This helps to monitor the position of the center of the robot. A PID controller can be used to help the Autobot move over the desired equation. If the Autobot deviates from the equation it was supposed to move over, an error signal would be generated. The constants Kp, Ki, Kd would be calculated by the micro-controller. If the Autobot, drifts to right of its desired path the PID controller would try to minimize the error by trying to move the Autobot in the left direction. In such a condition the micro-controller needs to vary the speed of the Left and Right motors in such a way that the Autobot drifts left till the center is not following the required equation of travel.

Drifting and turning the Autobot

A very gradual turn in the direction of Autobot by very small angle is meant to be a drift in the direction of motion. If the Autobot turns at larger angles then it is termed as turning the Autobot. A drift can be achieved by having a small variation in the speed of the left and right wheels. It is well known that, along a body, if two wheels are moved with different speeds then the body may turn gradually or suddenly depending upon the difference of speed of the wheels. If the body needs to be turned gradually to the left the speed of the left wheel needs to be reduced with respect to the right wheel. If the body needs to turn gradually to the right, then the speed of right wheel needs to be reduced with respect to the left wheel. If a major turn has to be made involving large angles, then the wheels need to be moved in opposite direction to each other. In such a case the micro-controller should calculate the co-ordinates of the center by taking the difference of the distances travelled by wheels. Varying the respective wheel speed of the Autobot, it can be turned in either direction.



Obstacle avoidance

Four cases can be discussed depending upon the position of the object.

Case I : Object Is Either On The Left Or Right Side Of The Line Of Propagation Of The Object: In such a case the obstacle is not sensed by the IR sensors mounted on the diametrical opposite ends. If the object lie to either the left or right side of the travel path of the Autobot, the Autobot will continue to move in the direction it was following, since the obstacles lie outside the line of propagation of the Autobot.

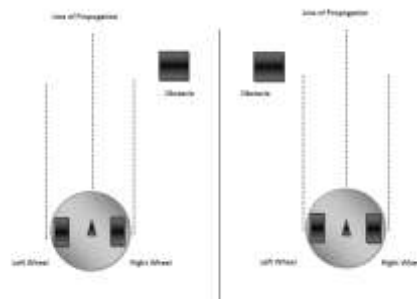


Figure 1: Position of Obstacle outside the Line of Propagation

Case II : Obstacle Lies between Left Wheel and Central Axis (Centre excluded):

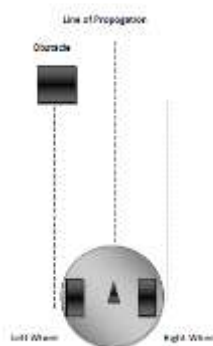


Figure 2: Position of obstacle between left wheel and central axis

If the obstacle is present along the left wheel of the Autobot, the Left IR sensor would sense its presence from a distance. In such a case, The Autobot would STOP at its position and the centrally mounted proximity sensor would sense the distance of the distance of the obstacle from it. The micro-controller present would calculate this distance say value “d”. Since, object is sensed by left IR, it will turn the Autobot by 90° to the right. And start moving forward in such a way that the distance between the centrally mounted proximity or



UV sensor stays constant i.e., “d”. In order to change the direction of motion the microcontroller would vary the speed of left and right wheels relative to each other such that the Autobot curves in a circular arc. The Autobot continue to curve, maintaining a constant distance “d”, until it returns to it predecided path of travel. The co-ordinates of the instantaneous position of the Autobot are continuously fed in the path of travel of the Autobot to check weather the Autobot has reached the original travel path. At the co-ordinates which satisfy the equation the Autobot would stop and calculate “ ϕ ” i.e., the angle between its current position and the next point it has to travel. The microcontroller would then vary the speeds of the wheels to achieve the desired turn in angle and the Autobot would continue to move in the respective direction of actual path.

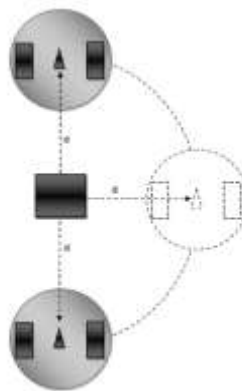


Figure 3: Path followed by the Autobot

Since, it senses the edge of the object, the Autobot would continue to move at a constant distance “d” from the edge of the object. As a result, it can decide its path such that it is always d distance away from the edge until it reaches the desired path of travel.

Case III : Object Lies between Right Wheel and Central Axis (Centre included):

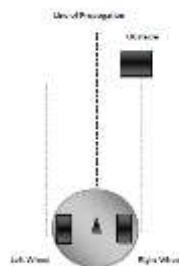


Figure 4 : Position of object between right wheel and central axis



Global Journal of Engineering Science and Research Management

The Autobot would follow a similar procedure as explained when the object was between the left wheel and central axis. The only difference is that it STOP and turn left by 90° and continue to follow a path at a distance “d” from the obstacle.

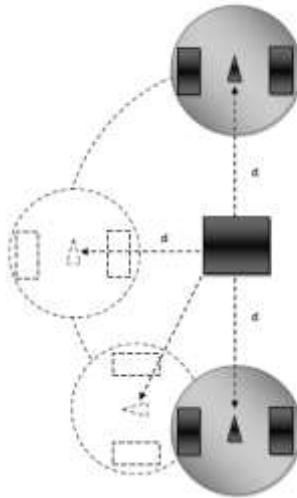


Figure 5 : Path Followed by the Autobot

Again, the Autobot would follow the path till it reaches the original equation of travel.

Tracing the Original Path

While travelling at a distance d from the object, the instantaneous position of the object is known in a virtual co-ordinate system. These co-ordinates are constantly fed into the original equation to check whether the Autobot has reached a point which existed in the original path of its travel. When the Autobot encounters such a point it STOPS and determines the next point of travel. When the Autobot reaches a point which lies on the original path of travel, there exist two new points on the original path towards which the Autobot can travel. Let them be P1 and P2. If the Autobot turns in the direction of P1 then, reaching at point P1 it needs to check the signal from its proximity sensor. If it has moved closer to the obstacle the output of the proximity sensor would increase and as a result, the Autobot would know that it has travelled towards the obstacle and thus, the path taken is wrong. In such a case, it has to move to the original point P0, and then move in direction towards P2 and then follow the desired equation of travel.

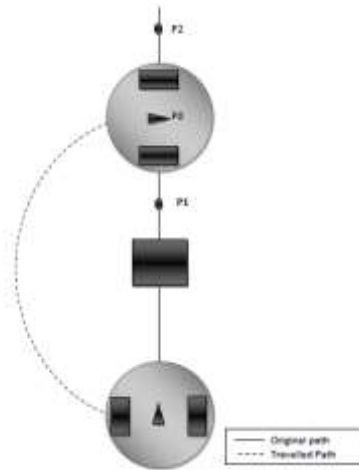


Figure 6 : Tracing the Original Path

Extending the approach to another obstacle encountered while negotiation the path from the first

If while negotiating its path, the Autobot experiences another obstacle, then it has to turn in the same direction which it had turned while encountering the first obstacle. It should turn by 90° in the direction it previously turned. If it turned left previously, and after moving some distance it encountered another obstacle then, it needs to STOP and measure the current distance between it and the new obstacle say “dnew” and turn further left by 90° and now move at a constant “dnew” distance from the new obstacle until it reaches the original path of travel. If another obstacle is sensed then, it should follow the same procedure, turning in the same direction as above. Hence, it can trace its path through more than one obstacle.

Limitations

It can be observed that there may be skidding between the tyres and the surface and hence, affect the wheel encoder readings. Speed of the Autobot needs to be limited to avoid skidding.

References

In the field of robotics the beginner can contribute many functional operations in the world. This arm can solve many human’s limitations. Many people cannot move from one place to another place for their limitation but they have needed to move for collect something like mug, jog, and so on. For that they require getting help from other persons. When they use this type of robot they can solve their problem easily without help other person for its easy operation system. For an example when a person has needed to carry an object from drawing room to bed room he can use this robot. It can move surround also collect photo and other information. When earthquake will be occurred by using these types of robot people can unseat many weight full objects from destroyed area to a safety place.



References

1. Barshan, B. & Durrant-White, H. F (1994) "Orientation Estimate for Mobile Robots Using Gyroscopic Information" *Proceedings of International Conference on Intelligent Robots and Systems (IROS '94). Munich, Germany, pp. 1867-1874.*
2. Borenstein, J. & Feng, L (1996) "Measurement and correction of systematic odometry errors in mobile robots" *IEEE Trans. Robot. Autom., Vol. 12, No. 6, pp. 869-880.*
3. Borenstein, J (1998) "Experimental results from internal odometry error correction with the omnimate mobile robot" *IEEE Trans. Robot. Autom., Vol. 14, No. 6, pp. 963-969.*
4. Box Gep & Hunter, W G (1978), *Statistics for experiments: An introduction to design data analysis and model building*, Wiley, New York.
5. Byrne, R.H., Kiarer, P.R. & Pletta, J.B (1992) "Techniques for Autonomous Navigation" *Sandia Report SAND92-0457, Sandia National Laboratories, Albuquerque, NM.*
6. Chaoli Wang, Zhenying Liang & Qingwei Jia (2010) "Dynamic feedback robust stabilization of nonholonomic mobile robots based on visual servoing" *Journal of Control Theory Applications, Vol.8, No.2, pp. 139-144.*
7. Chenavier, F. & Crowley, J (1992) "Position Estimation for a Mobile Robot Using Vision and Odometry" *Proceedings of IEEE International Conference on Robotics and Automation, Nice, France, pp. 2588-2593.*
8. Eghtesad, M. & Neculescu, D.S (2006) "Study of the internal dynamics of an autonomous mobile robot" *Robotics and Autonomous Systems, Vol.54, pp. 342-349.*
9. Evans, J. M (1994) "HELPMATE: An Autonomous Mobile Robot Courier for Hospitals" *International Conference on Intelligent Robots and Systems (IROS '94). Munich, Germany, pp. 1695-1700.*
10. Gonzalez-Gomez, J., Victores, G., Valero-Gomez, A. & Abderrahim, M (2011) "Motion Control of Differential Wheeled Robots with Joint Limit Constraints" *Proceedings of 6th International Symposium on Autonomous Mini robots for Research and Edutainment, pp. 165-172.*
11. Hollingum, J (1991) "Caterpillar make the earth move: automatically" *The Industrial Robot, Vol. 18, No. 2, pp. 15-18.*
12. Jianxian Cai & Xiaogang Ruan (2011) "Bionic autonomous learning control of a two-wheeled self-balancing flexible robot" *Journal of Control Theory Applications, Vol.9, No.4, pp. 521-528.*
13. Julian Lategahn, Frank Kuenemund & Christof Roehrig (2010) "Mobile Robot Localization using Wlan Odometry and Gyroscope data" *International Journal of Computing, Vol. 9, Issue 1, pp. 22-30.*
14. Kleeman, L (1995) "Odometry Error Covariance Estimation for two Wheel Robot Vehicles" *Technical Report MECSE-95-1, Intelligent Robotics Research Centre, Department of Electrical and Computer Systems Engineering, Monash University.*
15. Komoriya, K. & Oyama, E (1994) "Position estimation of a mobile robot using optical fiber gyroscope (OFG)" *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, Munich, Germany, pp. 143-149.*
16. Kooktae Lee, Changbae Jung & Woojin Chung (2011) "Accurate calibration of kinematic parameters for two wheel differential" *Journal of Mechanical Science and Technology, Vol.25 No.6, pp. 1603-1611.*
17. Korayem, M.H., Bani Rostam, T. & Nakhai, A (2006) "Design modelling and errors measurement of wheeled mobile robots" *International Journal of Advanced Manufacturing Technology, Vol.28, pp.403-416.*
18. Krantz, D (1996) "Non-Uniform Dead-Reckoning Position Estimate Updates", *Proceedings of IEEE International Conference on Robotics and Automation, Vol. 2, pp. 2061-2066.*
19. Lauro Ojeda & Johann Borenstein (2004) "Methods for the Reduction of Odometry Errors in Over-Constrained Mobile Robots" *Autonomous Robot, Vol. 16, pp 273-286.*



Global Journal of Engineering Science and Research Management

20. *MaciejTrojnacki (2012) "Analysis of Influence of rive system Configurations of a Four Wheeled Robot on its Mobility" Journal of Automation, Mobile Robotics & Intelligent Systems, Vol.6, No.4, pp. 65-70.*
21. *Montgomery, D.C (2000), Design and analysis of experiments, John Wiley and sons, New York.*
22. *ShenZhi-cua, zhoying-kai& Wu Wei-wei (2006) "Niche pseudo-parallel genetic algorithms for path optimization of autonomous mobile robot" Journal of Shanghai University, Vol.10, No.5, pp. 449-453.*
23. *Yao Cai, Qiang Zhan & Xi Xi (2012) "Path tracking control of a spherical mobile robot" Mechanism and Machine Theory, Vol.51, pp. 58-73.International Journal of Scientific and Research Publications, Volume 3, Issue 12, December 2013 9ISSN 2250-3153*