A Dynamically Adapting Indoor Navigation Algorithm for Mobile Robots

Zihao Yuan, Kevin Laubhan, Kumar Yelamarthi
School of Engineering and Technology
Central Michigan University
Mount Pleasant, MI 48859
k.yelamarthi@ieee.org

Abstract

Mobile robot has many applications especially in industrial field. One of the most challenging problems in mobile robot is the indoor navigation. This paper presents a dynamically adapting navigation algorithm for indoor mobile robots. A portable integrated system in an embedded system equipped with a Pioneer P3-DX robot with an array of sonar sensors is used. Based on the input received at the start of operation, the proposed algorithm can dynamically calculate a route, avoid obstacles, and travel to reach the predefined destination. The proposed algorithm avoids obstacles with assistance from ultrasonic sensors and also provides user feedback information about obstacles and direction changes through a feedback system. The prototype was successfully implemented and tested.

Introduction

Since the industrial revolution, machines have increasingly become more efficient and effective in assisting humans in a variety of tasks. In recent years, the advancements in robot technology have had a profound on the personal and professional lives of people around the world[1-3]. In industry, robots have become commonplace because of the difficult and often time consuming tasks they could handle with uniformity, speed, and precision. Though the intellectual ability of industrial robots has matured in certain applications, it has not grown equally in certain applications. For instance, there has been tremendous interest for application of navigation assistive robots to operate in unstructured environments. Broadly stating, applications of assistive mobile robots are endless in fields ranging from medical sciences[4], industry[5], and household tasks[6].

In this paper, we propose to model, design, and implement an integrated system that can be used for efficient localization and navigation in industrial applications. Goal of the proposed effort is to develop a portable integrated system for robot localization and navigation in indoor environments. Fig. 1 shows the proposed integrated platform. The method proposed by this paper utilizes eight forward facing ultrasonic (sonar) sensors in front of the robot to allow for maximum visibility. For mobile robots to be effective and efficient motion from the starting point to the final goal position needs to be a calculated trajectory based on the sensed objects and different possible paths. As the robot proceeds onto what it considers an optimal path, it will continuously monitor its environment, identify obstacles, makes an extensive map of the environment, and dynamically calculates the shortest route to the destination.

Previous Work

One of the fundamental challenges in mobile robots is navigation, which is the ability to find a safe path to travel from a starting position to the end goal while localizing itself in the environment using sensorial data. Given a map and a goal location, path planning involves finding a geometric path from the actual location of the robot to the goal/target. This type of planning is referred to as static planning, as the map is not updated dynamically per the obstacles or new information. Numerous research methods have been proposed in static planning such as Probabilistic Roadmaps[7], Rapidly Exploring Random Trees (RRT)[8], Generalized-Sampling Based Methods, and Visibility Graphs[9], Voronoi Digrams[10], and cell decomposition methods[11]. These methods are all different and all have positive and negative effects on the path of the robot and the project itself. Probabilistic roadmap is a method that accumulates data about the environment, creates random probability paths over time based on the relative distance of the object to the robot. It allows for the robot to traverse several different areas without exclusivity since it continuously analyzes and computes the probability of an object existence. The problem with this method is that it does not allow for direct and optimal time travel to reach the destination. On the other hand, the problem with RRT is that it does not allow for direct and optimal time travel to the destination. While it can analyze the environment efficiently, it fails at minimizing the time to reach end goal as its computational power is focused on calculating probabilities of success in travelling the random paths created. Overall, a problem in RRT is that it produces a path with many branches over times due to its natural behavior of using the randomized technique.

The second challenge in the mobile robots is obstacle avoidance. While some obstacles in the robot navigation path might be static, majority of the obstacles in real-world application are dynamic in nature. This requires the navigation algorithms to be dynamic in nature by constantly estimating its position, detect any obstacles, and update its route to the continuously changing trajectory to reach the destination. Considering the significance of obstacle avoidance in mobile robot navigation, several research methods such as bug algorithm[12], artificial potential field (APF) method[13], virtual force field method[14], vector field histogram (VFH)[15] method has been proposed. The bug algorithm constantly senses the environment for obstacles, and when it senses an obstacle, it moves around until it reaches the goal. One major problem with this method is that the trajectories generated by this method are often very long, and the robot it prone to moving too close to the obstacles. One other popular method proposed is the artificial potential field (APF) method, where the obstacles are represented by an attractive potential, and the robot traverse by avoiding the obstacles. The major problem in the APF method is finding the local minima, which causes a serious problem for the robot to reach its destination. The VFH method uses two-dimensional Cartesian histograms to calculate robot's current position, trajectory with low density of obstacles to reach its destination. However, the limitation with this method is that it does not consider nonholonomic constraints of the mobile robot. Answering these challenges, in this paper, a new dynamically adapting indoor navigation algorithm for mobile robots is proposed. Utilizing the on-board sonar array, the algorithm presented constantly monitors the environment, detects the presence, shape, and size of any static or dynamic obstacles, and dynamically adapts its trajectory to reach the destination without any challenges in local minima.

Fig 1: Mobile Robot used for indoor navigation

Obstacle Detection

The first step in obstacle avoidance is accurately identifying location of the obstacle with respect to the robot. With eight sonar sensors present at the front of the robot as in fig. 2, we are able to have a clear 180^o obstacle detection range. Based on the preliminary assumption that these eight sensors are evenly spaced, angle between any two consecutive sensors is computed as in eq.1.

$$q = (180 / n - 1) \qquad (1)$$

where, n is the number of sensors in the array. Accordingly, the angle between any two consecutive sensors was found to be 25.71^o. In addition, the angle of sensor-i with respect to the front of the robot is calculated as in eq. 2. This information would assist towards performing a coarse estimation on the location of the obstacle. When a obstacle is detected by each sensor, it outputs a distance ‘d’ as marked in fig.2. With each sensor having a unique obstacle detection angle, location of the obstacle can be detected with a fine accuracy through using basic trigonometry.

The first step in the process of obstacle detection is identifying the origin where edges of each sensor meet as marked by (x_o,y_o) in the figure. Once the coordinates of the origin are found, the distance between the origin and edge of each sensor is computed and marked by $c_0, c_1, c_2, \ldots c_7$. Once the distance between each sensor to origin has been found, coordinates to edge of the sensor can be found as in eq (3), where, *i* is number of the sonar. Once the robot starts navigating, the sonar array continuously looks for obstacles right across, and each sensor computes distance ($d_0, d_1, d_2 \ldots d_7$) to the obstacle. When an obstacle is detected by multiple sensors, the size and shape of the obstacle can be computed as in eq.(4), where (x_{obs-i}, y_{obs-i}) are the coordinates of the obstacle with respect to the origin as reported by $sonar_i$. Accordingly, distance from all eight sonars to the obstacle is computed to find coordinates (x_{obs-i}, y_{obs-i}). Once this information has been obtained, the shape of the obstacle can be found as in eqs (5)-(7).

$$\theta_i = 25.71 \times i \qquad (2)$$

$$(x_i, y_i) = (c_i \times \cos(\theta_i), c_i \times \sin(\theta_i)) \qquad (3)$$

$$c_i + d_i = \sqrt{(x_{obs-i} - x_o)^2 + (y_{obs-i} - y_o)^2} \qquad (4)$$

$$if\ (x_{obs-i} \text{ » } x_{obs-i+1} \text{ » } x_{obs-i+2}),\ Shape = Straight \qquad (5)$$

$$if\ (x_{obs-i} \text{ £ } x_{obs-i+1} \text{ £ } x_{obs-i+2}),\ Shape = Curved\ left \qquad (6)$$

$$if\ (x_{obs-i} \text{ ³ } x_{obs-i+1} \text{ ³ } x_{obs-i+2}),\ Shape = Curved\ right \qquad (7)$$

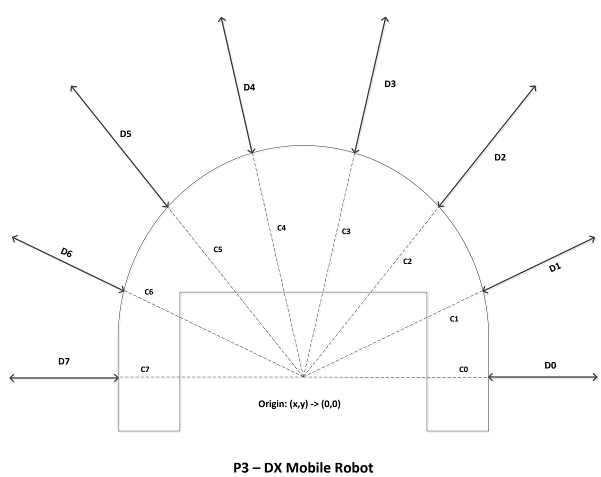

Fig. 2: Placement of sonars on front of robot

Navigation and Obstacle Avoidance

The robot requires specific instructions to effectively navigate indoors. While navigating, the robot must also avoid obstacles as detected by the sonars. The proposed method uses simple trigonometric functions to calculate a shortest path from the current location to its destination. By initializing the input as a coordinate point *(x, y)*, the robot moves towards its destination by continuously estimating its current location, and dynamically creating a shortest path as in fig. 3. With obstacle avoidance being of high significance in robot navigation, the first step in this algorithm is initializing safe distance to obstacle 'd_{obs}.' Once this information has been provided, the system asks for destination coordinates (x_d, y_d), and the robot estimates its current location (x_c, y_c) using dead reckoning method[16]. After the coordinate information has been obtained, the robot calculates the distance *'dd'* and angle (θ) it has to travel as in eq. (8) and eq. (9).

$$dd = \sqrt{\left(x_d - x_c\right)^2 + \left(y_d - y_c\right)^2} \qquad (8)$$

$$\theta = Tan^{-1}\left(y_d - y_c / x_d - x_c\right) \qquad (9)$$

Once the distance to destination and angle of rotation has been calculated, the robot rotates accordingly, and starts moving towards the destination. During this process, the robot continuously monitors its environment through sonars to detect any objects in its path. If any objects are detected within the distance 'd_{obs},'the robot terms it as an obstacle and initiates the obstacle avoidance subroutine. If the sonars detect no objects, the robot dynamically estimates its current location, calculates the new distance and angle of rotation, and reaches its destination with a high precision.

When the robot initiates the obstacle avoidance subroutine, it first determines which of the sonars as in fig.2 are detecting the obstacles. With this information, using eqs. (4)-(7), it calculates the size and location (x_{obs}, y_{obs}) of the obstacle. Next, in order to compute the new route to avoid obstacles, it checks if multiple sonars are detecting the obstacle. If only sonar is detecting the obstacle, it marks the number of the appropriate sonar as S_L. If multiple sonars are detecting the obstacle, it lists the left most sonar as S_L, and right most sonar as S_R. For instance, if sonars-1, 2, and 3 are detecting the obstacle, $S_L = S_3$, and $S_R = S_1$. Later, to calculate the angle of rotation (ϕ), first it calculates the angle between S_R and S_L as in eq. (10). Next, the relative location (left/right) of the obstacle is determined based on the value of S_L. If $S_L \geq 4$, obstacle is identified as being on front left side of the robot. If $S_L \leq 4$, obstacle is identified as being on front right side of the robot.

$$\phi = \left(S_L - S_R\right) \times 25.71^O \qquad (10)$$

Conclusion

The concept of using ultrasonic sensors for object detection and obstacle avoidance has been shown to be implementable. Through use of trigonometry, continuous sensor readings, and algorithms centered on using the most efficient routes possible to reach the destination. The proposed navigation algorithm is a viable solution to many personal and industrial needs. Future work involves extensive testing for localization using ultrasonic and RFID technology, accurate obstacle size and shape detection, and reducing computational requirement.

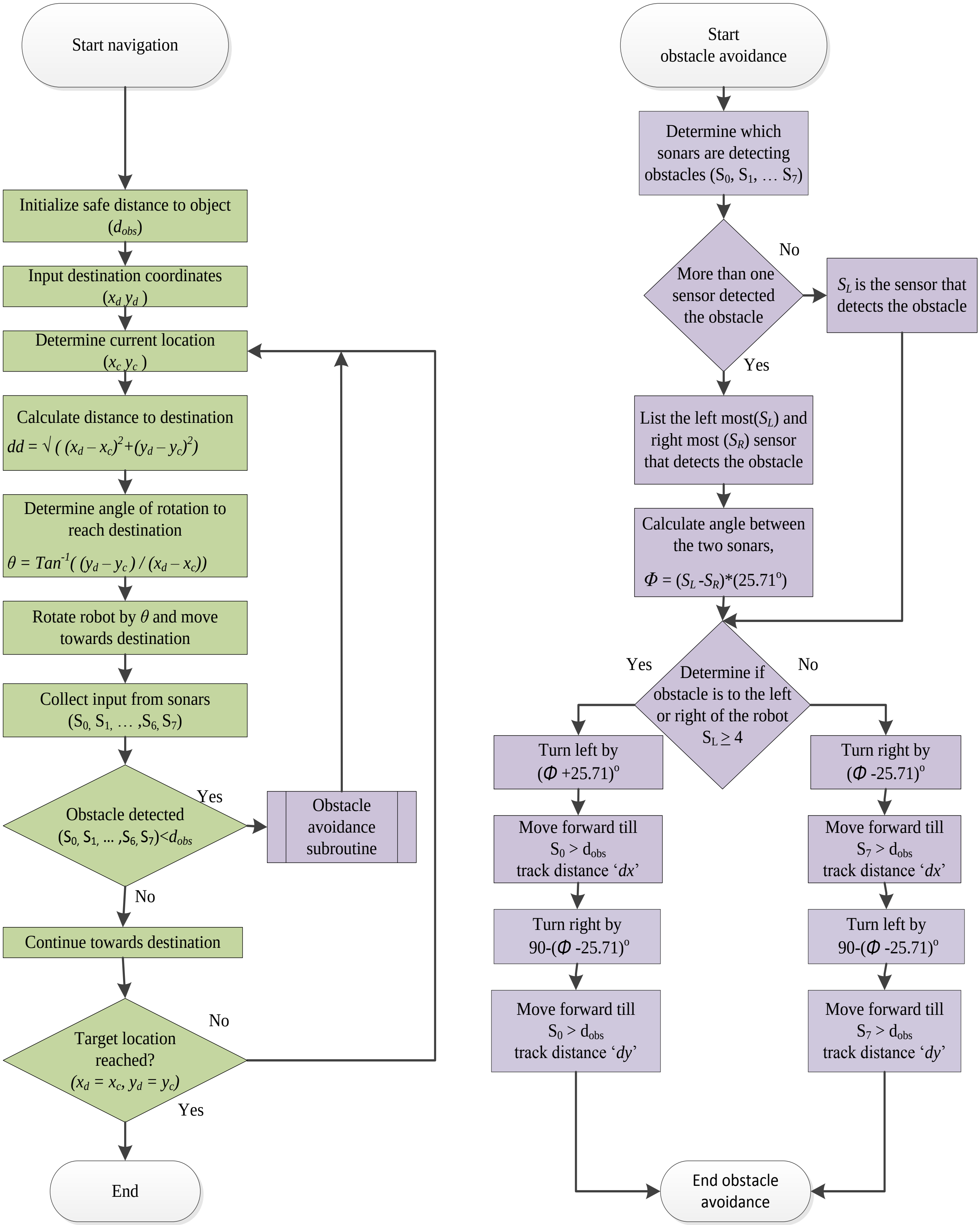

Fig 3: Navigation Algorithm & Obstacle Avoidance Subroutine for Mobile Robot

Bibliography

1. H. Cheng, He. Chen, Y. Liu, W. Sheng, "Human-like indoor navigation for Autonomous Industrial Mobile Manipulator," *IEEE International Conference on Information and Automation*, pp.162-167, June 2012.

A. Breitenmoser, R. Siegwart, "Surface reconstruction and path planning for industrial inspection with a climbing robot,"*International Conference on Applied Robotics for the Power Industry*, pp.22-27, Sep 2012.

2. K. Yelamarthi, S. Sherbrook, J. Beckwith, M. Williams, R. Lefief, "An RFID based Semi-Autonomous Indoor Tour Guide Robot," *IEEE International Midwest Symposium on Circuits and Systems*, pp. 562-565, Aug 2012.
3. X. Duan, Y. Wang, X. Kong, M. Li, Y. Yang, "Mechanical design and kinematic analysis of a medical robot assisted maxillofacial surgery," *International Conference on Complex Medical Engineering*, pp.596-601, May 2013.
4. J. Beckwith, S. Sherbrook, R. Lefief, M. Williams, K. Yelamarthi, "CATE: An Indoor Tour Guide Robot," *IEEE International Conference on Electro/Information Technology*, May 2012.
5. Z. Xiao, M. Su, Y. Guan H. Zhang, Q. Yan, Z. Wang, "Wawa: A tumbler-like household robot," *IEEE International Conference on Mechatronics and Automation*, p.623, 628, 5-8 Aug. 2012.
6. L.E. Kavraki, P. Svestka, J.C. Latombe, M.H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566-580, Aug 1996.
7. S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," *CS Dept., Iowa State Univ.*, Ames, Tech. Rep. 11, 1998.
8. S. Chakravorty, S. Kumar, "Generalized Sampling-Based Motion Planners," *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, vol. 41, no.3, pp. 855-866, June 2011.
9. F. Aurenhammer, "Voronoi diagrams–a survey of a fundamental geometric data structure," *ACM Computing Surveys*, vol. 23, no. 3, pp. 345–405, Sep 1991.
10. R. Siegwart, I.R. Nourbakhsh, "Introduction to Autonomous Mobile Robots," *MIT Press*, 2004.
11. V. Lumelsky and A. Stepanov, "Autonomous Robot Vehicles," *Springer-Verlag*, pp.363 -390, 1990.
12. O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Proceedings of IEEE International Conference on Robotics and Automation*, vol.2, pp.500-505, Mar 1985.
13. J. Borenstein, Y. Koren, "Real-time obstacle avoidance for fast mobile robots," *IEEE Transactions on Systems, Man and Cybernetics*, vol.19, no.5, pp.1179-1187, Oct 1989.
14. F. Xu, H. Van Brussel, M. Nuttin, R. Moreas, "Concepts for dynamic obstacle avoidance and their extended application in underground navigation," *Robotics and Autonomous Systems*, vol. 42, issue. 1, pp. 1-15, Jan 2003.
15. W. Chen, R. Chen, Y. Chen, H. Kuusniemi, J. Wang, "An effective Pedestrian Dead Reckoning algorithm using a unified heading error model," *IEEE Position Location and Navigation Symposium*, pp.340-347, 4-6 May 2010.