FINAL PROJECT:

OBSTACLE AVOIDANCE CHALLENGE PREPROPOSAL

MAE 412 Mobile Robotics

Department of Mechanical and Aerospace Engineering

Benjamin M. Statler College of Engineering and Mineral Resources

West Virginia University

Submitted: Wednesday September 25, 2019

Submitted To: Dr. Yu Gu

Submitted By: Hayden L. Nichols

# Technical Discussion

## Introduction and Objective

The obstacle avoidance challenge is a competition in which a fully autonomous robot must explore and area, avoiding collisions with stationary obstacles and other robots in the competition area. This document provides a review of relevant literature and outlines a proposed approach to the competition.

The main objective of the mission is to obtain points, defined by distance travelled and number of collisions. Additional points are given to teams that explore all areas of the field. The robot must operate entirely without human intervention and must remain productive for the entirety of each of the three 12-minute-duration trials.

The SMART robot will be the platform used to develop and test the obstacle avoidance system. It consists of a 2-wheeled Roomba chassis, a 9-axis IMU, wheel encoders, a 2D LIDAR rangefinder, and capabilities to utilize ultra-wide band (UWB) ranging.

A survey of existing approaches and relevant algorithms, techniques, and system designs is reviewed. Among these relevant articles, topics discussed relate to obstacle detection and prediction of mobile obstacles, representation of obstacles and exploration goals in useful formats (e.g., occupancy grids, vector field histogram, etc.), decision making and path planning, and control laws for mobile robots.

The outlined approach to the task and relevant research has been focused on specific subdivisions of the task: obstacle detection and tracking, path planning and avoidance, robot motion and control, and integration and testing. Aspects of the proposed approach relating to each of these subdivisions, as well as a secondary contingency plan for development are presented and discussed.

## Literature Review

Obstacle avoidance can be viewed as a dichotomy: obstacle detection and avoidance path planning. In general, there are two approaches to obstacle detection and avoidance: one based on a global map, and one based entirely on sensors oriented about the robot’s local frame [8]. Given the possibility for reconfiguration of the environment and the dynamic nature of the environment, a global map approach may not be feasible. As such, articles detailing a local, sensor-based approach will be emphasized.

For the case of obstacle detection using the SMART robot, information about the obstacles in the robot’s environment must be gleaned from 2D LIDAR range data. One method for producing information about obstacles from 2D LIDAR in the robot local frame involves a pipeline of point cloud filtering to remove noise and reduce the complexity of clustering, segmentation and merging segments, and clustering nearby segments into obstacles. This method reports size of the obstacle and position relative to the robot [8].

Characteristics of the obstacles in addition to size and relative position, such as relative velocity of the obstacle, can also be tracked. One approach that expands obstacle detection to non-stationary obstacles uses a Kalman-based process to predict velocities of detected obstacles [1]. Tracking obstacle velocity can allow for decision-making in a dynamic environment. This can be applied thorough a Bayesian approach to an occupancy grid style reactive algorithm [1,2]

Once obstacles have been detected, the avoidance path planning or reaction algorithm can be applied. One of the earliest, and perhaps simplest, algorithms is the Bug algorithm. Bug plans a direct path to the destination until it faces an obstacle. There are three varieties of the bug algorithm, based on intermediate behavior when avoiding the obstacle [9]. The bug algorithm requires a distance sensor and knowledge of the current and destination position. Although it may move the robot away from the destination or take a low-efficiency path, it is low-computation and always converges [5].

Another low-computation algorithm is the artificial potential field (APF) method, which simulates a potential field by assigning repulsive potential to obstacles and attractive potential to goals. The net artificial force of these potentials on the robot are calculated based on the strength of the potential field and the robot position in the field. This net force results in acceleration actuated by the robot control system. AFP is goal-oriented and can generate the shortest path, but can also generate dead-end solutions or local minima when symmetric or U-shaped obstacles are encountered [6, 10]. Moreover, APF does not take into account robot constraints [5], such as non-holonomic constraints on a mobile robot, therefore it may be better suited for manipulator and UAV planning than for mobile robots.

A more efficient and tunable avoidance planning algorithm is the follow the gap method (FGM), which determines the maximum gap between obstacles is calculated and the center angle of the gap from the robot frame is the bearing angle for the path [7]. FGM always chooses the shortest path and does not have problems with symmetric obstacles. The calculations are simple to compute and the only needed information is obstacle distance and angles. However, the solution may not be reached with dead-end obstacles [5].

A more computationally complex method of avoidance that is well-suited to non-holonomic mobile robots is the vector field histogram (VFH) method. This projects obstacles into a 2D histogram representing occupies cells and associated certainties. The 2D cartesian histogram is converted into a 1D histogram representing the polar densities of obstacles surrounding the robot. From this 1D polar histogram, the direction associated with a region whose density is below a set threshold is chosen as the goal heading, and a linear velocity is set in this direction [3]. Unfortunately, the generation of the histograms can be computationally costly, but for the case of a 2D LIDAR, this is less concerning. A method that works similarly, but avoids local traps and has been applied to two-wheeled differential drive is the obstacle restriction method (ORM), which also takes into account proximity of a potential path to the path toward the goal location [4].

## Proposed Approach

### Obstacle Detection and Tracking

The proposed approach will utilize the 2D LIDAR onboard the SMART robot to detect objects surrounding the robot. These points will be segmented to detect planes (walls) and clustered to group points into obstacles. Then, the VFH algorithm will be performed.

### Path Planning and Avoidance

Global path planning will be accomplished by partitioning the competition area into regions. The robot will begin with an initial guess of which region it has started in and another region will be selected as a goal region. The initial guess may be aided by use of the UWB measurement. As the robot traverses toward the goal region, it will utilize either dead reckoning or an extended Kalman filter to track position relative to the initial guess. As new information about the position in the environment (i.e., a planar surface or a corner is detected), the planner will optimize the guessed start position and relative path to include this new information. As the goal region is reached, the goal region will be changed to a different region of the competition area, attempting to maximize exploration. Since total distance traversed is the main goal, accuracy of the understanding of the environment is not completely necessary. If the robot position has not changed significantly over a given period of time, the goal will be changed to act as a perturbation to move the robot from a trap.

Avoidance will be performed using the VFH method, as it easily lends itself to application to 2D LIDAR. However, the basic VFH method will be extended to account for mobile obstacles using Kalman-based observer discussed in [1]. The static obstacles will be mapped into the polar histogram and the projected paths of the Kalman-tracked mobile obstacles (assuming constant velocity) will be propagated through time to generate a trajectory, which will be included as an obstacle in the histogram. The avoidance trajectory will be determined based on this histogram. An additional approach that may be implemented for negotiation of moving obstacles is planning in the velocity space by estimating times when parts of the histogram will be occupied and adjusting the robot speed accordingly, similar to [1].

### Robot Motion and Control

The robot control will utilize the fact that the robot can drive both forward and backwards without adjusting heading. Since the goal set by the global path planning is only to generate motion, moving toward the goal will not be prioritized, so driving backwards away from approaching or impassable obstacles is a valid and productive option. The nearest clear heading from the VFH will be chosen and the velocity will be determined based on the presence of moving obstacles. A PID controller on heading may be implemented, depending on results of preliminary testing.

### Integration and Testing

All avoidance and planning will be done inside the SMART robot SMART\_run.m script loop, so that the robot’s sensors are integrated. This script will be modified to read sensors and send motor commands only when necessary, to maintain processing speed.

### Secondary Approach

A secondary approach will be developed in the event of failure of the primary approach, or to support the primary approach using a decision process. This secondary approach will utilize the same detection of obstacles but will ignore the motion of obstacles and rely on a Bug1 algorithm. The inefficiencies of Bug1 may serve to benefit the performance in this challenge, since it is not goal-based, but scored on distance travelled and exploration.

## Related Experience

My related experience in robotics includes 2+ years working in robotic research, focusing in robot perception and autonomy and a summer internship working in GNSS-denied precision navigation of ground vehicles. Additionally, I am currently in my second year of leading autonomous systems development for WVU Mountaineer Robotics Team for University Rover Challenge.

# References

[1] A. Cherubini, F. Spindler and F. Chaumette, "Autonomous Visual Navigation and Laser-Based Moving Obstacle Avoidance," in IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 5, pp. 2101-2110, Oct. 2014.  
doi: 10.1109/TITS.2014.2308977

[2] C. Fulgenzi, A. Spalanzani and C. Laugier, "Dynamic Obstacle Avoidance in uncertain environment combining PVOs and Occupancy Grid," Proceedings 2007 IEEE International Conference on Robotics and Automation, Roma, 2007, pp. 1610-1616. doi: 10.1109/ROBOT.2007.363554

[3] J. Borenstein and Y. Koren, “The vector field histogram-fast obstacle avoidance for mobile robots,” IEEE Transactions on Robotics and Automation, vol. 7, no. 3, pp. 278–288, 1991.

[4] J. Minguez, "The obstacle-restriction method for robot obstacle avoidance in difficult environments," 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Alta., 2005, pp. 2284-2290.  
doi: 10.1109/IROS.2005.1545546

[5] M. Zohaib and M. Pasha and R. A. Riaz and N. Javaid and M. Ilahi and R. D. Khan, “Control Strategies for Mobile Robot With Obstacle Avoidance,” 2013.

[6] Oroko, J., & Ikua, B. (2012). Obstacle Avoidance and Path Planning Schemes for Autonomous Navigation of a Mobile Robot: A Review. Sustainable Research and Innovation Proceedings, 4.

[7] Sezer, V., & Gokasan, M. (2012). A novel obstacle avoidance algorithm:“Follow the Gap Method”. Robotics and Autonomous Systems

[8] Y. Peng, D. Qu, Y. Zhong, S. Xie, J. Luo and J. Gu, "The obstacle detection and obstacle avoidance algorithm based on 2-D lidar," 2015 IEEE International Conference on Information and Automation, Lijiang, 2015, pp. 1648-1653.  
doi: 10.1109/ICInfA.2015.7279550

[9] Yufka, A., & Parlaktuna, O. (2009, May). Performance comparison of bug algorithms for mobile robots. In Proceedings of the 5th international advanced technologies symposium, Karabuk, Turkey.

[10] Zhu, Y., Zhang, T., Song, J., & Li, X. (2012). A new hybrid navigation algorithm for mobile robots in environments with incomplete knowledge. Knowledge-Based Systems, 27, 302-313