ECE498: Senior Capstone Project I

Project Proposal

Project Title: Mobile Target Tracking Using Radio Sensor Network

Nicolas Auth and Grant Hovey
Advisor: Dr. Suruz Miah

Electrical and Computer Engineering Department
Caterpillar College of Engineering and Technology
Bradley University

Table of Contents

Abstract

One of the most fundamental problems in the field of mobile robotics is the ability for a mobile robot to both localize itself and generate a map in an unknown environment using only external sensor measurements. There is a great deal of research that has been conducted on this subject using many different approaches and methodologies. There is however still work that needs to be done in this area, as there can always be a more precise or efficient solution. A particular challenge in the simultaneous localization and mapping area is that of a cost effective and modular solution that can be used in an indoor environment. Even more complexity can be added to this challenge if one extends the scope to include solving the mobile target tracking problem as well. The mobile target racking problem is as such; the mobile target and follower robot are both placed in an unknown environment where neither the pose (position and orientation) of either are known a priori. The follower must then match the unknown trajectory of the mobile target using only external sensory measurements. This work approaches this problem using range only measurements generated by a network of wireless sensors placed randomly throughout the robot's operating environment. A custom built radio transceiver is mounted on the follower robot to gather these measurements. The path of the mobile target in the operating environment is completely random and represented by another radio sensor in the wireless sensor network. A conventional Extended Kalman Filter Simultaneous Localization and Mapping (EKF-SLAM) algorithm will be used to process the wireless measurements taken by the radio receiver. The EKF-SLAM algorithm will estimate the pose of the follower robot, the position of the mobile target, and generate a map of the environment by estimating the position of the wireless sensors placed in the operating environment. A PI (proportional integral) controller will be used to calculate the necessary velocity for the follower robot to track the mobile target within a minimum safe distance. We will first simulate this work using a commercial robotics simulator V-REP (Virtual Robot Experimentation Platform) and then implement it using a mixture of custom and pre-built hardware.

1 Introduction

Mobile robots are quickly becoming a ubiquitous facet of modern life due to their applications in industry with heavy equipment trucks, in agriculture for autonomous crop maintenance, in home life with the autonomous vacuuming robot, along with research and design as seen in [1, 2, 3, 4]. Tracking a mobile target (location is unknown) by a mobile robot is one of the fascinating problems as it can be applied in many applications, such as in [5]. A large body of research has been conducted in the literature to address the target tracking problem. However, in most cases, either (i) target tracking algorithm is analyzed and its performance is evaluated using computer simulations, (ii) static target is considered for a robot to track it as seen in [6], (iii) assumes that the target's location is known, or (iv) expensive hardware platforms are used to implement the mobile target tracking strategy like in [7].

The current project addresses some of the aforementioned issues by implementing a cost-effective and easy-to-implement mobile target tacking scheme using a differential-drive mobile robot and a set of networked wireless radio sensors. The design and implementation of the current work is based on the preliminary work of the robot navigation and mapping strategy conducted by the authors in [4, 8, 9]. Therefore, the main objective is to advance the work in [4, 8, 9] for a more general case, where a mobile robot is supposed to track a mobile target whose position is a priori unknown.

2 Background Study

In this project, it is emphasized that the location of the mobile target is unknown. The proposed target tracking system has to simultaneously estimate the target's position and the pose (position and orientation) of the mobile robot that is supposed to track the target while building the map of the environment. Therefore, a major part of the current project is to address the simultaneous localization and mapping problem for the mobile robot operating in an indoor environment. There is a depth and breadth of robotics research focusing on both the simultaneous localization and mapping problem, as well as the mobile target tracking problem (see [10], [11], [12], and some references therein). A good bit of this research is now focused on the use of passive RFID tags and signal strength readings. In the research done in [10] and [11], UHF (ultra high frequency) RFID is used for the localization of a mobile robot. The signal strength and phase information from the RFID tag's signal was used to calculate the angle towards and distance from the robot to the RFID tag. Localization was not fully proven in these experiments as the robot followed a programmed path. The main issue with this methodology was the intensive calculations needed to extract the bearing of the RFID tag from the phase shift of the signal.

In another work as seen in [12], an algorithm with a similar principle is implemented using wireless radio sensors as beacons instead of RFID tags. This system is much easier to implement, scale-able, and more cost effective. The environment is sectioned in to a grid map to facilitate easy calculation of signal strength from the wireless beacons that is converted in to distance and bearing measurements. The wireless signals were captured using a fixed forward mounted antenna.

One research project not directly related to localization and mapping [13] shows a method for obtaining angle and distance measurements from a radio signal. A rotating receiver dish is used to capture wireless signals from multiple sources. Distance is calculated using the signal strength, and bearing is calculated using the angle of the receiver dish. This method was accurate to within $\pm 3°$.

Using the radio based EKF-SLAM system from the previous year's project, we aim to extend the simultaneous localization and mapping solution to include mobile target tracking. This will be accomplished using a rotating receiver dish as in the previously mentioned research project [13] as well as a Velodyne puck LiDAR sensor. One of the radio beacons in the network will be identified as the mobile target and will be tracked using both the signal captured by the receiver dish as well as the Velodyne LiDAR.

3 Standards and Technology

There are several wireless communication standards that apply to this project including the ZigBee radio protocol and IEE 802.15.4. The IEEE 802.15.4 protocol[1] describes specifications for the medium access control layer (MAC) and the the physical layer (PHY) for implementing low data rate wireless communication in a personal area network (LR-WPAN). The MAC specifications detail access to the physical channel for transmission and defines the IEEE device specific address. The PHY specifications details the radio frequency transceiver and the low-level controller. This protocol is low cost, easy to use, and reliable with low energy usage for extended battery life. The beacon network and mobile target will use XBee modules to communicate with the mobile robot using the 802.15.4 wireless protocol.

The Zigbee standard [23] is an open standard based on IEEE 802.15.4 for low data rate wireless communication at low cost using minimal power. It uses the same PHY and MAC layer from 802.15.4 while in addition specifying a network topology layer and application framework. The network layer defines the necessary routing, security, and structure for establishing networked communication. The application framework layer defines addressing objects and device specific applications. The Zigbee network layer allows for several topologies including peer-to-peer, mesh, and star. Our radio sensor network uses a star topology with the mobile robot's Zigbee radio serving as the central coordinator. This coordinator establishes the network assigning unique IDs for each of the other Zigbee radios in the network.

[1] IEEE Standard for Low-Rate Wireless Networks, 2015
[2] Zigbee Pro Stack User Guide,2014
[3] XBee/XBee-Pro S2C ZigBee User Guide, 2016

A Velodyne VLP-16 Puck LiDAR sensor will also be used in this work. The Puck is the smallest LiDAR sensor produced by Velodyne and is capable of real time 360° 3D distance measurements. This sensor has a measurement range of 100 meters with a power consumption of 8 watts making it ideal for operating on a battery powered robot in an indoor environment.

4 Functional Requirements

The goal of this project is to track a moving target using a mobile robot in an unknown environment. This target tracking will be accomplished using a localization and mapping algorithm implemented using the mobile robot's on-board sensors as well as a network of radio sensors. This algorithm estimates the position and orientation of the mobile robot as well as the location of the radio units in the sensor network and the location of the moving target.

To be more specific we will implement the tracking of a moving target by a differential drive mobile robot in an indoor environment. We will be using a Pioneer P3-DX as our mobile robot. This tracking will first be accomplished using an extension of the Extended Kalman Filter Simultaneous Localization and Mapping (EKF-SLAM) algorithm. The robot will estimate its position and orientation, the position of several static radio sensors, as well as a radio sensor serving as the moving target by measuring the signal strength. A BeagleBone Black microcomputer will serve as the controller for the mobile robot as well as receiving signal strength and ID from the radio sensors and target. We will use ZigBee radio modules as our radio sensors mounted at three-dimensional (3D) coordinates around our indoor environment. A general look at the virtual environment for this project is shown in Figure 1.

The EKF-SLAM algorithm will be implemented on the microcomputer. This will estimate the pose (position and orientation) of the mobile robot, the 3D coordinates of the radio sensors, and the coordinates of the moving target. A control algorithm will then be used to determine the appropriate actuator commands to the differential drive mobile robot's left and right wheels to move it towards the moving target based on their estimated pose and distance in a 2D plane.

After this basic moving target tracking is accomplished, we will attempt to make the moving target tracking even more accurate by incorporating a Velodyne LiDAR sensor. LiDAR stands for Light Detection And Ranging. A LiDAR sensor uses pulses of light to measure distance to a target. The LiDAR sensor will provide an estimate of the moving target's position with greater precision than the SLAM algorithm and reduce the target tracking error. As a further extension, we may also implement a basic obstacle avoidance algorithm using the robot's on-board sonar sensors.

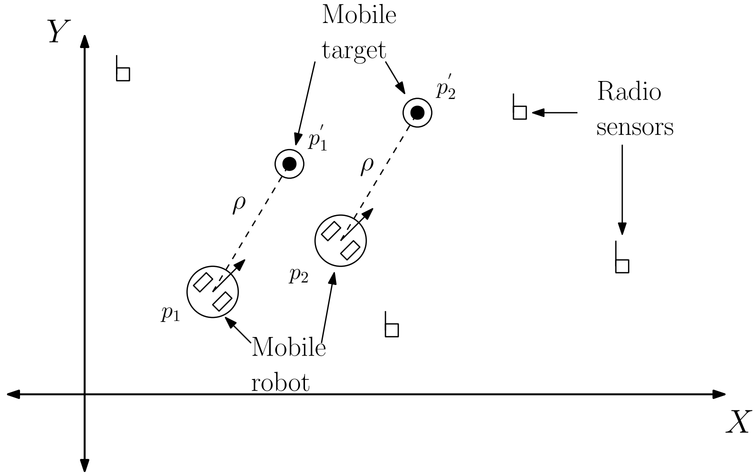


Figure 1: Environment with mobile robot, ceiling mounted radio sensors, and mobile target.

5 System Architecture

The high level system architecture of our project is shown in Figure 1, where p_1 and p_2 represent the mobile robot's position at time instances 1 and 2, and $p_1^{'}$ and $p_2^{'}$ represent the mobile target's position. The high level system block diagram is shown in Figure 2. The mobile target tracking system accepts the trajectory of the mobile target as its input and then generates and outputs a tracking performance metric. This performance metric is calculated using the difference between the estimated trajectory of the mobile target and the position of the mobile robot and which the tracking system is implemented.

5.1 Block Diagrams

The subsystem level diagram shown in Figure 3 is a closer look at what makes up the system. The subsystem takes the trajectory of a moving target as an input and outputs a tracing performance metric. The estimated trajectory of the moving target is fed into a tracking algorithm. This tracking algorithm provides the actuator commands to the mobile robot that will send it towards the moving target. These actuator commands are calculated based on the moving target trajectory, the position of the robot, and a map of the environment. The mobile robot also outputs the tracking metric based on its position relative to the mobile target. A feedback loop is implemented using a series of sensors. This feedback loop will attempt to minimize the trajectory performance metric by improving the trajectory estimate sent to the tracking algorithm.

The *estimated pose* and *estimated positions of XBees* are used internally. This feedback design allows the system to use previous data gathered and use the predictions to estimate the mobile robot's position with greater accuracy the longer it is running. The internal values output by the *Tracking algorithm* block are the linear

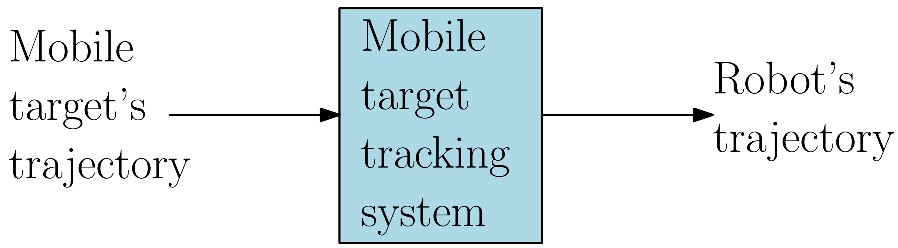


Figure 2: System level block diagram.

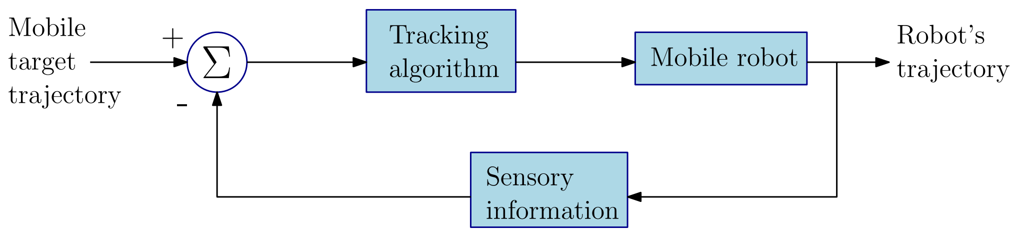


Figure 3: Subsystem level block diagram.

velocity and the steering angle. These values are then converted to control inputs accepted by the Pioneer P3-DX. Once these control inputs are applied and the mobile robot has completed its movement, the noisy pose is sent to the estimation block where it is used to estimate the mobile robot's true pose. Mapping of the XBee radio sensors within the environment is also done. The estimated pose is then used by the controller in order to repeat the process.

5.2 Specifications

For the proposed project, there are four main specifications that should be met. First, the overall cost should be less than \$500 in order to stay within the guidelines set by the department. Secondly, the mobile robot must be able to be localized within 30 cm of its true position. Thirdly, the position estimates of each beacon should be within 30 cm of its true position with respect to the indoor operating space of the robot. Finally, the tracking performance of the moving target to be within 30 cm.

6 Preliminary Work

The following sub-sections will cover the work done so far. This includes the modelling of the mobile robot, system simulation results, and the current and future design work.

6.1 Modelling

Since the algorithm for EKF-SLAM is widely-used and well known, we did not need to create the model from scratch. This model is only used for calculating the steering angle of the mobile robot with pose $\mathbf{q}^r = [x^{[r]}, y^{[r]}, \theta^{[r]}]^T$ where $\mathbf{p} = (x^{[r]}, y^{[r]})$ is the robot's 2D position and $\theta^{[r]}$ is its orientation with respect to the global coordinate

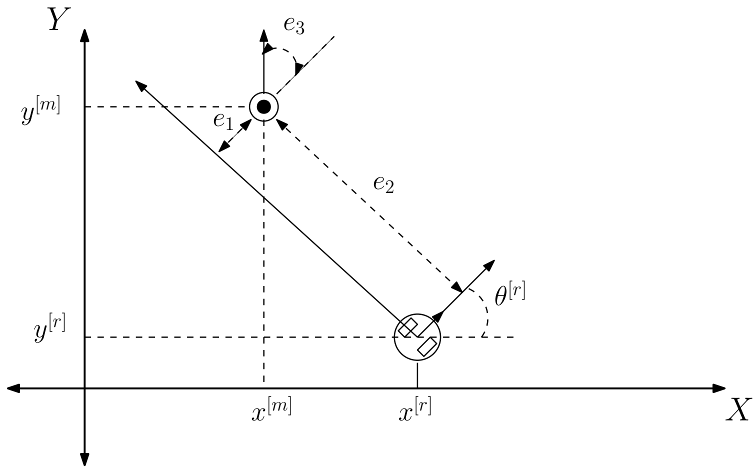


Figure 4: Tracking error diagram.

system. This model will calculate the necessary steering angle to converge toward the moving target with location $\mathbf{q}^m = [x^{[m]}, y^{[m]}, \theta^{[m]}]^T$ where $\mathbf{p}' = (x^{[m]}, y^{[m]})$ is the mobile target's 2D position. In this model the linear velocity is calculated using a PI controller that is shown by

$$\nu(i) = K_p d_T(i) + K_i \int_0^i d_T(\tau) d\tau \tag{1}$$

where $\nu(i)$ is the mobile robot's linear velocity during iteration i, $d_T(i)$ is the normalized distance between $\mathbf{p}$ and $\mathbf{p}'$, and K_p and K_i are the proportional and integral constants respectively.

The steering angle is calculated using

$$\gamma = \gamma + \Delta\gamma \tag{2}$$

where $\Delta\gamma$ is the change in the steering angle calculated by

$$\Delta\gamma = \tan^{-1} \frac{y^{[m]} - y^{[r]}}{x^{[m]} - x^{[r]}} - \theta^{[r]} - \gamma \tag{3}$$

and the tracking errors of the mobile robot in terms of its local coordinate frame are described using

$$\mathbf{e}(t) = \begin{bmatrix} e_1(t) \\ e_2(t) \\ e_3(t) \end{bmatrix} = \begin{bmatrix} \cos\theta(t) & \sin\theta(t) & 0 \\ -\sin\theta(t) & \cos\theta(t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^{[m]}(t) - x^{[r]}(t) \\ y^{[m]}(t) - y^{[r]}(t) \\ \theta^{[m]}(t) - \theta^{[r]}(t) \end{bmatrix} \tag{4}$$

An illustration of the relationship between these angles and the positions can be seen in Figure 4.

Since a differential drive mobile robot is being used in the project, the steering angle γ must be converted to a usable value, in this case angular velocity, ω. The angular velocity is calculated by

$$\omega = \nu \frac{\sin \gamma}{\ell} \tag{5}$$

where ℓ is the distance between the wheels.

6.2 Simulation Results

For the simulation of our algorithm, we chose to use the free, multi-platform robot simulator V-REP[4]. The V-REP robot simulator follows industry standard robotic platforms so the results are consistent in a way that is accurate to their real-world counterparts. This simulator is popular within the robotics community due its integrated development environment, large number of robot models available, the ability to write in a variety of programming languages including C/C++ and Matlab, its compatibility with ROS, and overall ease of use.

The model presented in section 6.1 was then implemented in V-REP. The theoretical results were gathered using the provided Pioneer 3-DX model for the simulation. Using V-REP's remote API, the Pioneer model is able to be controlled through MATLAB. Figure 5 shows the environment for the simulation.

Since the algorithm operates under the assumption that there will be noise in the environment, we added Gaussian noise in MATLAB. In our simulation, noise is introduced as process noise and observation noise, each with a noise covariance matrix. The process noise covariance matrix is represented by

$$\mathbf{Q} = \begin{bmatrix} \sigma_\nu^2 & 0 \\ 0 & \sigma_\omega^2 \end{bmatrix} \tag{6}$$

where σ_ν is the standard deviation of linear velocity calculated by $\sqrt{\frac{1}{T^2} 2 (\frac{1\pi r}{3N})^2}$ where r is the radius of the wheels, T is the time step, and N is the number of encoder ticks per wheel revolution. For the Pioneer we use the following values: $r = 0.075\,\text{m}$, $T = 0.25\,\text{s}$, and $N = 500$ which gave us the calculated value of σ_ν to be 8.8858×10^{-4}. σ_ω is the standard deviation of angular velocity. For this simulation we chose a value of $\sigma_\omega = 0.0012\,\text{rad}$.

The observation noise covariance is represented by

$$\mathbf{R} = \begin{bmatrix} \sigma_R & 0 \\ 0 & \sigma_B \end{bmatrix} \tag{7}$$

where σ_R is the standard deviation of the range measurement and σ_B is the standard deviation of the bearing measurement. We chose $\sigma_R = 1\,\text{m}$ and $\sigma_B = 0.0873\,\text{rad}$

[4] http://www.coppeliarobotics.com/

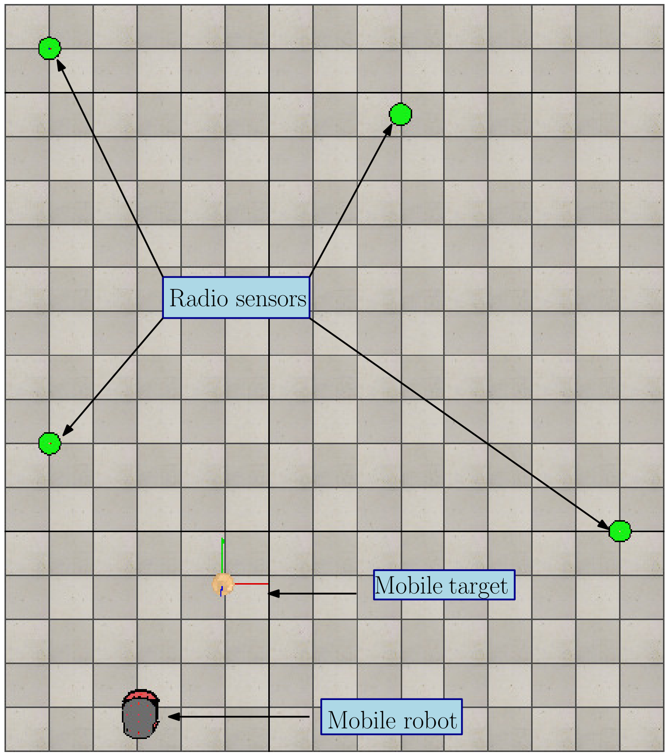


Figure 5: Scene in V-Rep of mobile robot, mobile target, and the radio sensors.

for the simulation.

Using these values, the simulation was run for 500 iterations. The simulation can be seen in Matlab in Figure 6. The simulation can be seen in V-Rep in Figure 7.

The error for the robot is defined as

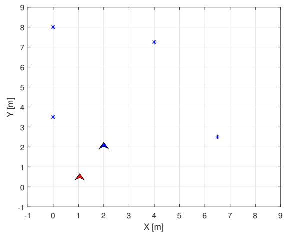
$$e_P(t) = \sqrt{(\hat{x}(t) - x^d(t))^2 + (\hat{y}(t) - y^d(t))^2} \tag{8}$$

where $\hat{x}(t)$ and $\hat{y}(t)$ are the estimated position values and $x^d(t)$ and $y^d(t)$ are the desired position values. Similar to (8), the error for the radio sensors can be calculated using

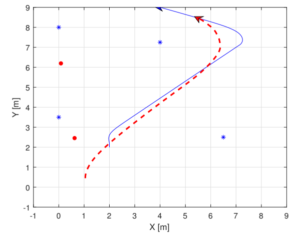
$$e_B(t) = \Sigma_{i=1}^{n} \sqrt{(\hat{x}_i(t) - x_i^d(t))^2 + (\hat{y}_i(t) - y_i^d(t))^2} \tag{9}$$

where n is the number of radio sensors, $\hat{x}_i(t)$ and $\hat{y}_i(t)$ are the estimated position of the ith radio sensor and $x_i^d(t)$ and $y_i^d(t)$ are the true position of the ith radio sensor.

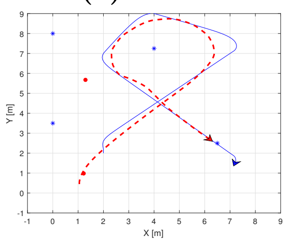
Using (10), the RMSE of the robot's error was calculated to be 0.0928 m. Using (11), the RMSE of the radio sensor error was calculated to be 1.7254 m. This number is deceiving since the combined error of all radio sensors was used, instead of a single radio sensor's error.



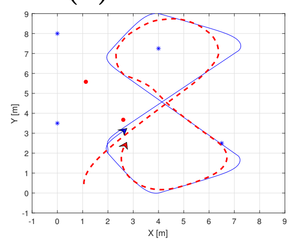
(a) $t = 0s$.



(b) $t = 25s$.

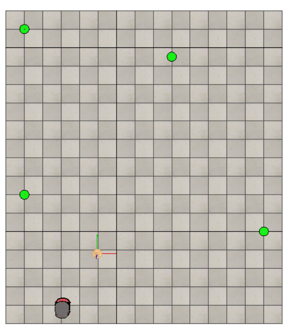


(c) $t = 50s$.

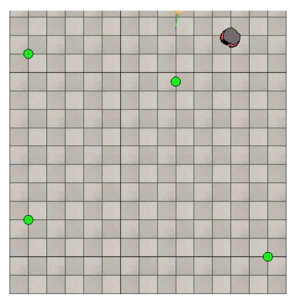


(d) $t = 75s$.

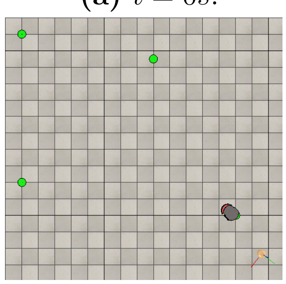
Figure 6: Matlab plots of mobile robot (red), mobile target (blue), radio sensor estimation (red star), and radio sensor actual (blue star) from $t = 0s$ to $t = 75s$.



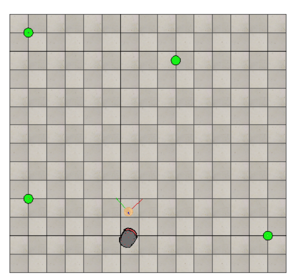
(a) $t = 0s$.



(b) $t = 25s$.



(c) $t = 50s$.



(d) $t = 75s$.

Figure 7: V-Rep plots of mobile robot chasing mobile target from $t = 0s$ to $t = 75s$.

Part	Price	Quantity
XBee S2C w/ whip antenna	$18.19	6
XBee Interface Board	$5.31	6
Perforated Circuit Board	$2.84	4
Stepper Motor	$14.95	1
Motor Controller	$19.95	1
3.3 V Regulator	$0.79	10
9 V Battery Clip	$0.39	10

Table 1: List of ordered parts

$$\mathrm{RMSE}_P = \sqrt{\frac{1}{t_f}\int_0^{t_f} [e_P(t)]^2 dt} \tag{10}$$

$$\mathrm{RMSE}_B = \sqrt{\frac{1}{t_f}\int_0^{t_f} [e_B(t)]^2 dt} \tag{11}$$

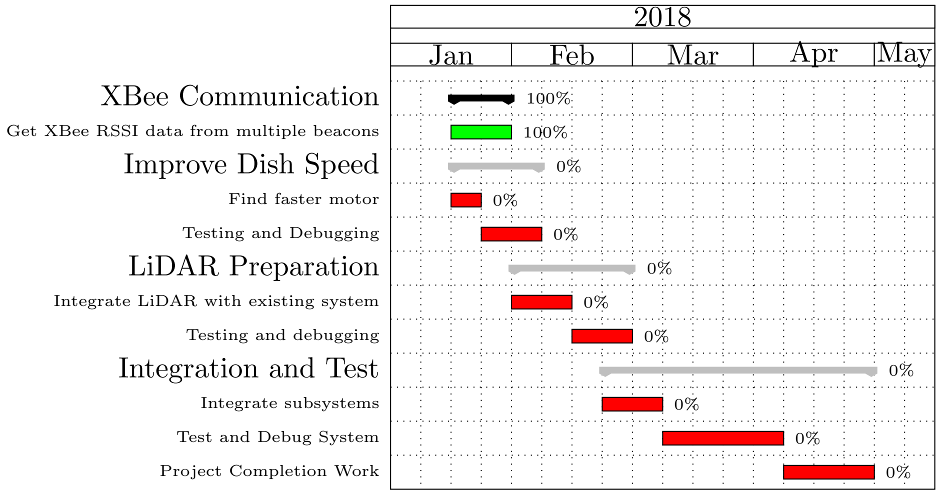
7 Parts List

The following parts list shown in Table 1 is from the previous year's project with the inclusion of the XBee interface board. We may require additional parts to increase the efficiency of the radio receiver but this has yet to be determined. Not included is the Pioneer P3-DX mobile robot platform or the Velodyne Puck LiDAR provided by the department.

8 Future Tasks

We will need to improve the radio sensor position estimation accuracy. We will need to become familiar with the LiDAR puck for implementing in the simulation as well as the physical implementation.

9 Timeline



References

[1] J. Roberts, A. Tews, C. Pradalier, and K. Usher, "Autonomous hot metal carrier - navigation and manipulation with a 20 tonne industrial vehicle," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, April 2007, pp. 2770–2771.

[2] H. Durmu, E. O. Gne, M. Krc, and B. B. stnda, "The design of general purpose autonomous agricultural mobile-robot: Agrobot," in *2015 Fourth International Conference on Agro-Geoinformatics (Agro-geoinformatics)*, July 2015, pp. 49–53.

[3] M. R. Elara, N. Rojas, and A. Chua, "Design principles for robot inclusive spaces: A case study with roomba," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 5593–5599.

[4] K. Hevrdejs and J. Knoll, "Indoor robot localization and mapping using zigbee radio technology," Bradley University, Department of Electrical and Computer Engineering, Tech. Rep., May 2017.

[5] E. Xu, Z. Ding, and S. Dasgupta, "Target tracking and mobile sensor navigation in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 1, pp. 177–186, Jan 2013.

[6] T. Fernando, H. Gammulle, and C. Walgampaya, "Fuzzy logic based mobile robot target tracking in dynamic hostile environment," in *2015 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, June 2015, pp. 1–6.

[7] V. Malyavej, W. Kumkeaw, and M. Aorpimai, "Mobile target localization and tracking," in *Wireless Sensor Networks. Signals and Communication Technology*, May 2014, pp. 1–6.

[8] J. Knoll, K. Hevrdejs, and S. Miah, “Virtual robot experiments for navigation in structured environments,” in *The 26th International Symposium on Industrial Electronics*, Edinburgh, Scotland, June 2017.

[9] K. Hevrdejs, J. Knoll, and S. Miah, “A zigbee–based framework for approximating sensor range and bearing,” in *The 30th annual IEEE Canadian Conference on Electrical and Computer Engineering*, Windsor, Ontario, Canada, April 2017.

[10] F. Martinelli, “A robot localization system combining rssi and phase shift in uhf-rfid signals,” *IEEE Transactions on Control Systems Technology*, vol. 23, no. 5, pp. 1782–1796, Sept 2015.

[11] E. DiGiampaolo and F. Martinelli, “Mobile robot localization using the phase of passive uhf rfid signals,” *IEEE Transactions on Industrial Electronics*, vol. 61, no. 1, pp. 365–376, Jan 2014.

[12] D. Song, C. Y. Kim, and J. Yi, “Simultaneous localization of multiple unknown and transient radio sources using a mobile robot,” *IEEE Transactions on Robotics*, vol. 28, no. 3, pp. 668–680, June 2012.

[13] B. N. Hood and P. Barooah, “Estimating doa from radio-frequency rssi measurements using an actuated reflector,” *IEEE Sensors Journal*, vol. 11, no. 2, pp. 413–417, Feb 2011.