INDIAN INSTITUTE OF TECHNLOGY, KANPUR

SURGE-2019

Range sensor based Localization and control of mobile robots

Author:
Naveen Balaji
Undergraduate Student

Supervisor:
Dr. Mangal Kothari
Associate Professor

End-Term Report for the project done in the

Intelligent Guidance and Control Laboratory
Aerospace Department
IIT Kanpur

December 3, 2020

Abstract

Unmanned Aerial Vehicles (UAVs) have the potential to assist us in real-world problems. Complex challenges like Transportation, Aerial Survey can be solved by the use of Swarm Robotics. In this project, We focus on new technologies in UAVs and their Algorithms that allow individual members of the swarm to communicate, plan, and coordinate their flight efficiently. We surveyed the existing method for Aerial Swarms, its computation, and cost required. We implemented swarm in drones using Real-time kinematic (RTK) GPS. We aim to build a system which can work on indoor areas and heavy industries where GPS estimates are not available. Our method is to use the Ultra-Wide Band (UWB) sensors for localization of UAVs. We have compared the different Gaussian filters like Recursive least square, Kalman and Information Filter. These algorithms are used to estimate position by fusing this sensor with the accelerometer, then evaluated the sensor accuracy both in static and dynamic mode. After this relative positioning, we explored some of the decentralized Flocking algorithms that will help the swarm to navigate and perform some tasks like shape formation.

Keywords:- Unmanned Aerial Vehicles, Gaussian filters, Swarm Robotics, GPS denied Navigation

.

Contents

Chapter 1

Introduction

Autonomous aerial systems capable of sensing and perceiving the environment have been the area of intense research due to their limitless applications, ranging from surveillance, precision agriculture, infrastructure inspection, photography, recreation etc. Unmanned Aerial Systems (UAS), such as quadcopters and unmanned helicopters, offer robust maneuverability with vertical takeoff and hovering capabilities on the three-dimensional airspace. They can perform tracking **kim2008moving** inspections **nigam2011control** and transportation **raptopoulos2016transportation** more quickly, economically, and safely compared to other comparable robots. The deployment of sophisticated sensors is incrementally enhancing the intelligence of these aerial robots **cai2014survey** enabling them to achieve autonomous navigation in complex and confined environments. The applications related to inspection and surveillance of commercial installations require these UAS to operate in GPS shadow areas where GPS signal reception may be diminished and less reliable. Further, the reliance of magnetometer for heading estimate is severely compromised if the UAS has to operate near large iron structures such as large cranes due to magnetic deviation.

Indoor Positioning System (IPS) becomes critical for many autonomous operations requiring application of UAS in GPS denied environments. Visual odometry for mobile robotics using feature tracking based on monocular and stereo-vision has been an active area of research to address indoor localization **scaramuzza2011visual**; **achtelik2011onboard**; **achtelik2009stereo** However, the solutions are sensitive to ambient lighting conditions, motion blur, and other artifacts that deteriorate image quality. Visual-inertial methods improved localization precision by eliminating scale factor error in the image with the fusion of inertial measurement unit (IMU) data **delmerico2018benchmark** The accumulation of drift in these methods are typical over time, thus cannot be used for long duration flights. Other SLAM based approaches popular among ground robots either use RGB-D cameras **kerl2013dense** or 3D lidars **xu2018slam** However, these approaches are often computationally expensive and time-intensive. They typically require onboard GPU computing ability to process the data in real-time, which increases the power requirements and the overall weight of the system, thereby further compromising the endurance of aerial vehicle.

The wireless localization system, such as Radio Frequency Identification (RFID) **zhou2009rfid** Wi-Fi **polo2014semantic** Zigbee **watthanawisuth2014design** and Ultra-Wideband (UWB) **alarifi2016ultra** are emerging technologies for indoor localization solutions. Due to the unsatisfied accuracy of the received signal strength (RSS) techniques **ruiz2011accurate** they have not been found suitable for UAVs. Recently, UWB based technologies have gained momentum in this field. With the large bandwidth, this signal has the properties of strong multi-path resistance, which enables accurate ranging via communication by the two-way time of flight. They are low-cost, low-power, portable, robust and easy to implement in any environment.

The present work is an extension of the approach followed in **gaur2018low** which focused on developing a pose estimation framework for a quadcopter relying on MARG (inertial) sensor array, an optical flow camera, and an Ultra-wideband (UWB) range sensor to correct the drift of the estimator over time. In this paper, use of multiple UWB sensors is proposed along with inertial sensors without any optical cameras for localization. Although some earlier research has focused on implementing Bayesian filter based state estimation using UWB sensors **mueller2015fusing**; **li2018accurate** it did not focus on optimizing the performance. The method used in this research aims to reduce the complexity of the algorithm using the extended information filter. The Gaussian filters require a model of the system, comprising of a state function, measurement function, and the associated noise terms. The noise terms related to it are often difficult to estimate. The inaccurate noise model can cause perturbation in the estimation, which will lead to divergence of the filter. There are optimal ways to adapt a filter according to the need **yang2006optimal** Noise covariance can be estimated by minimizing the cost function, with the known ground truth of the vehicle. In this paper, various criteria for tuning the filter are discussed, and the Particle Swarm Optimization (PSO) is used for determining the best noise covariance.

Real-time tracking of the heading angle in rigid bodies has wide applications in robotics fields **barshan1995inertial** Inertial and magnetic sensor modules with their associated data filtering algorithms are designed for estimating the attitude of the object **mahony2008nonlinear** The famous estimation algorithms such as **madgwick2010efficient** allows accurate evaluation of pitch and roll attitude but are not robust for yaw estimations over time. The sources of magnetic interference are always present in common items such as current-conducting wires, batteries, and ferrous materials. Today there are many hybrid solutions such as **bentley2016wireless** with expensive multiple sensors, to be used in the industrial environment for the estimation of the heading. This paper proposes to solve the problem of yaw estimation through a novel low-cost yaw estimation method without drift which can be used on UAS as fail-safe in the event of deterioration in yaw estimates from conventional MEMS magnetometers.

The remainder of the paper is structured as follows: the background work and problem statement is summarized in Sec. II. The localization and heading algorithms are given in Sec. III. The method of tuning the noise parameters based on PSI is given in Sec.IV. The experimental results and discussion are provided in Sec. V and concluding remarks are given in Sec. VI.

Chapter 2

Localization

Robot localization is the process of determining where a mobile robot is concerning its environment. All mobile robot irrespective of the work they do, they first need to understand where they are located. After knowing their current location, they can decide and perform any future actions instructed to them. Apart from basics navigation, mobile robots also needed position feedback for their controller to function. Today we came across a lot of improvement in this part due to the development of different types of low-cost electronic sensors. We will categorize the sensors based on the usability in the environment. Then we will see their estimation accuracy, computation, and their scalable factors in real-world projects.

2.1 Sensors

2.1.1 Global Possitioning System (GPS)

GPS is a satellite navigation system that gives location and time information. The principle of GPS is based on time difference taken by the signals to travel from satellite and sensors, with the help of approximated speed of the electromagnetic wave in the atmosphere we can calculate the distance between satellite and sensor. The exact location is determined by the trilateration method, minimum there should be at least four satellites. Minimum three satellites are used to trace the location and fourth one used to confirm the position.

2.1.2 Real Time Kinematic (RTK-GPS)

As the GPS signal travels through the Earth's atmosphere, it gets delayed. Many factors like climate, electromagnetic fields can increase position error in GPS. We assume that these factors do not change much in one area. In

RTK-GPS two receivers are used, one is a stationary called base station, another one is a rover fixed on the vehicle. Both the GPS sensors undergo the same deflection from the atmosphere; the base station measures errors and transmits corrections to the rover.

We have experimented of localizing the drones with the help of GPS and RTK GPS and have differentiated the results, the massive improvement in the RTK algorithm was observed

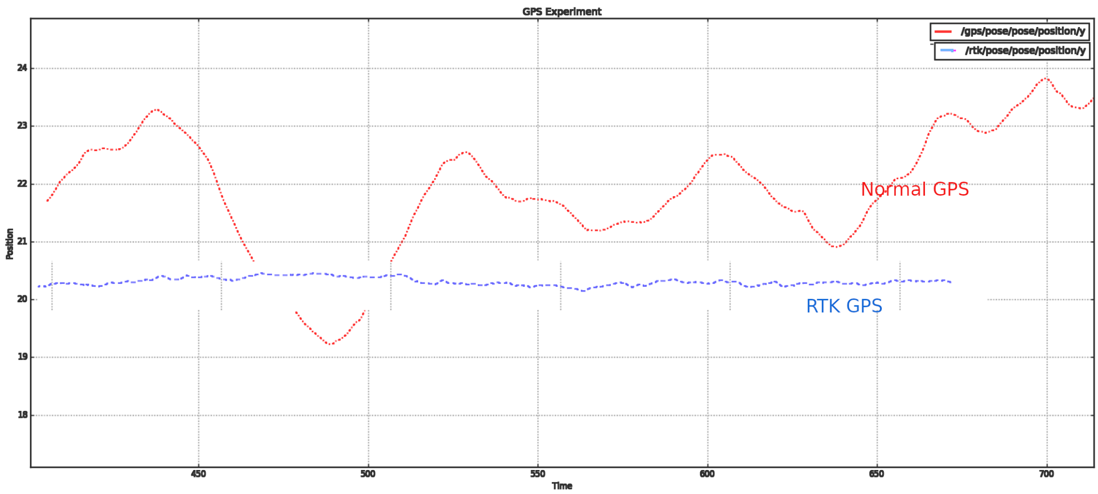


FIGURE 2.1: Location of RTK-GPS vs Normal GPS in a Static position Test

Position Error	Normal GPS	RTK GPS
Mean	1.2m	0.08m
Maximum	3.4m	0.15m

TABLE 2.1: Accuracies of above plotted graph

2.1.3 Vision Based:

It is the computer vision task of retrieving the pose of the camera given a query image of it. This method involves using an optical sensor like monocular camera, stereo camera, depth camera, Lidar laser, Optical flow sensor, etc., The workflow of most of the algorithms are common first process is *environmental perception* which will detect image features. By the model of sense they will apply localizing algorithms which will predict its location. There is huge growth in Map-building and map interpretation methods, but they need a lot of computation and features. Some navigation system is a part of SLAM (*simultaneous localization and mapping*), which can even reconstruct the environment. As our problem is for finding localization solution on any environment, we will not concentrate on these Map-based localization techniques. On the other hand, Visual data are grouped into local, global and hybrid features. Based on feature representation algorithms are applied.

- **Global Features:** Advances on Neural Networks helped us to localize in large scale estimates. Visual Place Recognition is one of the methods used for recognizing the environment by the robot with CNN.
- **Local Features:** It will depend on pixel level differences in consecutive images. Picks selected features in image like Edges, Geometry, Color of the environment. Some of the famous features to track are SIFT, SURF, ORB. Recently localization using monocular cameras is effective using the method. Solution s like ROVIO, VINS-Mono, MSCKF, SVO vision techniques are compared here[].

2.1.4 other techniques:

Improvement in embedded system has led to many new type of sensor in the market. They are not exploited much compared to above mentioned sensors.

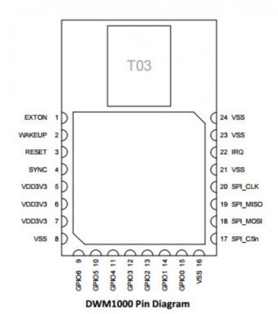
Received Signal Strength (RSS) – This method uses wireless sensors for signal transmission. RSS method relies on a path loss model, where the distance between receiver and transmitter is inferred by measuring the energy of the received signal. Not reliable, as some times it involve discontinous change of location. Wifi, Bluetooth are generally used communication in this method.

Angle of Arrival (AOA) – The AoA method measures the angle between the node and the incoming signal using typically multiple (array) antennas. By measuring the angle to three transmitters in 3D, the location of the receiver is found at the intersection of the three lines. Needed noise less environment such as underwater place, and complex instrument for precise angle estimation.

Time of Arrival (TOA) – The distance between the moving target and the fixed anchors is proportional to time taken by the signal to transmit. This is one way communication,and all the devices in the system should be synchronized. Signals like IR(*InfraRed*), Ultra Wide Band (UWB), Ultrasonic sound are sended between devices.

2.2 Ultra-WideBand sensor

The Ultre Wideband (UWB) technology has been the subject of extensive research in last two decades. It has emerged as promising candidate for many wireless applications, sensor networks and minimum computation.



(B) Pin Diagram

(A) sensor model

FIGURE 2.2: Ultra-WideBand Sensor–Decawave dwm1001

This is mainly due to its large system bandwidth (occupies more than 5 GHz of spectrum) which offers high accuracies with low-power/cost implementation.The signals are extremely short pulses with low power spectral density.

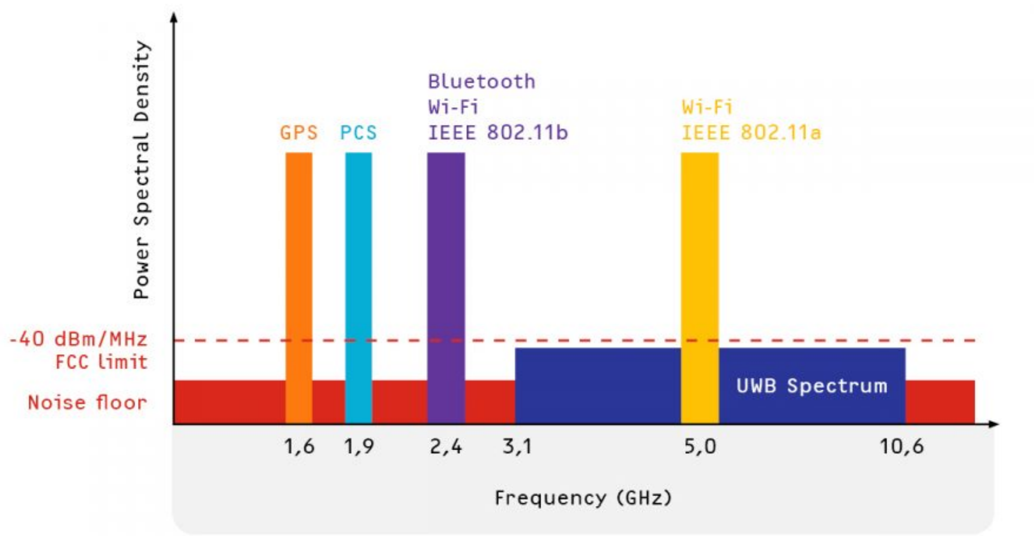


FIGURE 2.3: Frequency plot of Ultra Wideband

2.2.1 Working Principle:

Ultra-wideband (UWB) is a radio technology based on the IEEE 802.15.4a and 802.15.4z standards that can enable the very accurate measure of the Time of Flight (ToF) of the radio signal, leading to centimeter accuracy distance/location measurement, range upto 200m.

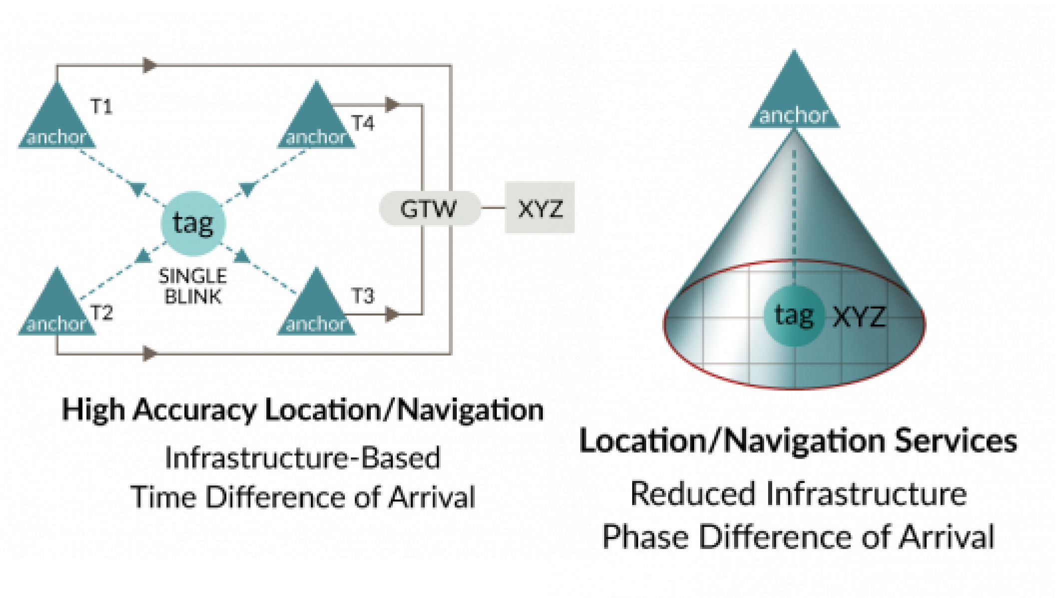


FIGURE 2.4: Working of Ultra-WideBand Sensor (dwm1001)

Time Difference of Arrival – The TDoA method is very similar to GPS. Multiple reference points, called anchors, are deployed in a venue and are time synchronized. The mobile devices will beacon, and when an anchor receives the beacon signal it will timestamp it. The timestamps from multiple anchors are then sent back to a central location engine which will run multilateration algorithms based on Time Difference of Arrival of the beacons signals to compute the X, Y, Z of the mobile devices.

Phase Difference of Arrival (AOA) –The PDoA method consists of combining the TWR scheme that delivers the distance between two devices with the measure of the bearing between the two devices. The combination of distance and bearing allows the calculation of the relative position of two devices without any other infrastructure. To do so, one of the devices carries two antennas and is able to measure the Phase Difference of Arrival of the RF signal.

Chapter 3

State Estimation Algorithms

We have seen a lot of sensors, each can vary for purpose and environment of use. Each sensor will provide bias, The bias is dependent on temperature, environment and several other factors that cause it to slowly change over time. These noise depend on the sensor sensitivity and calibration, generally expensive sensors are prone to environmental changes and vibrations. Noise causes issues to state estimator, we can't rely on a single sensor or estimation technique. We will explore family of Gaussian filters algorithm and use the best to fit our purpose.

3.1 Least Square regression

The method of least squares is about estimating parameters by minimizing the squared discrepancies between observed data.This method is computationally convenient technique to fit data. It corresponds to maximum likelihood estimation when the noise is normally distributed.
Let x be variable which sensor is measuring (eg: current) and y be Output variable shown by the sensor by some linear function (eg:voltage)

$$\text{Linear System Model} \quad y = Hx + v$$

v is the some sensor noise After getting many observations we want fit this system and estimate a sensor measurement $\hat{\boldsymbol{x}}$.

$$\text{The error term will be} \quad \varepsilon = y - H\hat{\boldsymbol{x}}$$

We will define a cost function for this and try to reduce this error as possible and fit it in our equation.

$$\begin{aligned} cost function \quad J &= \varepsilon_1^2 + \varepsilon_2^2 + ... + \varepsilon_n^2 \\ J &= (y - H \times \hat{\boldsymbol{x}})^T (y - H \times \hat{\boldsymbol{x}}) \end{aligned}$$

In order to minimize J with respect to $\hat{x}$

$$\frac{\partial J}{\partial \hat{x}} = 0$$
$$2\, y^T\, H + 2\, \hat{x}^T\, H^T\, H = 0$$
$$\text{Solving for } \hat{x}, \qquad \hat{x} = (H^T H)^{-1} H^T y$$

There are a lot of modified versions other than this basic model, We will see how to apply this method in our Position Estimation using Ultra Wideband sensor Now consider the above figure. Let A, B, C, D be the fixed

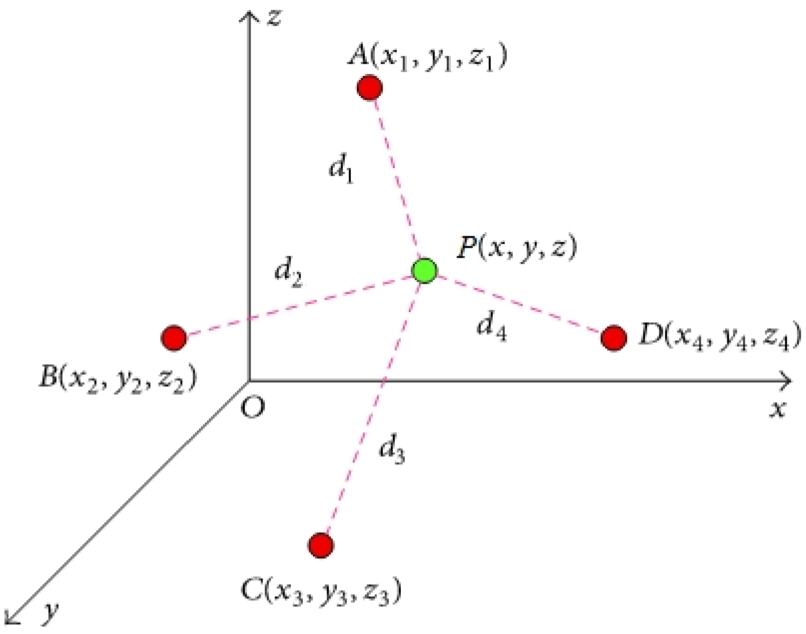


FIGURE 3.1: Setup of Working Environment

anchors with respect to a frame of reference xyz (fixed) whose values are known.P for the vehicle point which we estimate by the sensor which provides distance d1, d2, d3, d4.

$$(P - A)^2 = d_1^2 \quad ...(1) \qquad (P - B)^2 = d_2^2 \quad ...(2)$$
$$(P - C)^2 = d_3^2 \quad ...(3) \qquad (P - D)^2 = d_4^2 \quad ...(4)$$

Subtracting (2), (3), and (4) from (1) we get:

$$2\ (B - A)\,.P = d_1^2 - d_2^2 - B^2 - A^2$$
$$2\ (C - A)\,.P = d_1^2 - d_3^2 - C^2 - A^2$$
$$2\ (D - A)\,.P = d_1^2 - d_4^2 - D^2 - A^2$$

$P = (x, y, z)$ $A = (x_1, y_1, z_1)$ $B = (x_2, y_2, z_2)$ $C = (x_3, y_3, z_3)$ $D = (x_4, y_4, z_4)$

$$2 \times \begin{bmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \\ x_4 - x_1 & y_4 - y_1 & z_4 - z_1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} d_1^2 - d_2^2 - |B|^2 - |A|^2 \\ d_1^2 - d_3^2 - |C|^2 - |A|^2 \\ d_1^2 - d_4^2 - |D|^2 - |A|^2 \end{bmatrix}$$

We obtain the $y = Hx$ form. If we solve exactly some times there will be no solution, We can use least square to obtain the estimted $\hat{x}$. This time let's apply Recursive least square method as sensor is dynamic,we will modify the equations to reduce the big matrix and to be robust.

3.2 Kalman Filter

The Kalman filter operates by propagating the mean and covariance of the state through time. One remarkable aspect of the Kalman filter is that it is optimal in several different senses. The filter is constructed as a mean squared error minimiser, but they are recursive and at the same time carry previous state to reduce uncertainies and noise in the new measurements. The detailed derivation of kalman filter is on **??**.We will see its implemetation in our model. Let us give you a idea of Kalman Filter and see its implementation with our sensors.

State Space – The state of a dynamic system is the smallest vector that summarises the past of the system completely. Knowledge of the state allow prediction of future dynamics and outputs the deterministic system in absence of noise. In our problem let us assume (x_k) is the state vector of the process at time (k).

Calculate Noise – To perform calculation despite measurement noise. The state is represented as a mean (μ) and variances (σ^2) of the normal distribution. The variance indicates the confidence level of the value. From previous section, to minimise the cost function (J), we model the system error using Gaussians distributions. $\mathbf{P}_0 = E\left[(\mathbf{x} - \hat{\mathbf{x}}_0)(\mathbf{x} - \hat{\mathbf{x}}_0)^T\right]$

The Recursive steps of the kalman filter are

- Known Variables are $\hat{x}_k$-State ,u_k-Control ,P_k-Covariance y_k-Measurement.
- State Prediction $\mathbf{x}_k = F.\hat{\mathbf{x}}_{k-1} + G.\mathbf{u_k}$
- Measurement step $\mathbf{y}_k = H.\mathbf{x}_k$
- Calculate Kalman gain $\mathbf{K}_k = \mathbf{P}_{k-1}\mathbf{H}_k^T \left(\mathbf{H}_k\mathbf{P}_{k-1}\mathbf{H}_k^T + \mathbf{R}_k\right)^{-1}$
- Update the State $\hat{\mathbf{x}}_k = \mathbf{x}_k + \mathbf{K}_k\left(\mathbf{y}_k - \mathbf{H}_k\mathbf{x}_{k-1}\right)$

3.2.1 Extended Kalman Filter-EKF

Many practical systems have non-linear state update or measurement equations. The EKF handles nonlinearity by linearizing the system at the point of the current estimate by the taylor series, and then the linear Kalman filter is used to filter this linearized system. It was one of the very first techniques used for nonlinear problems, and it remains the most common technique.

Coming to our Problem, We are getting the range from anchors, and our quadcopter is equiped with Inertial Measurement Unit (accelerometer + gyroscope + magnetometer). Let us form our System equation and measurement model based on this information. And then apply the Kalman Filter to it.Directly integrating the acceleration from IMU lead to worse results, so we also consider the acceleration bias(a_b) in every step.
In the EKF, the state vector consists of position,velocity and acceleration bias in all the three directions. $\mathbf{x} = [p_x, v_x, a_{bx}, p_y, v_y, a_{by}, p_z, v_z, a_{bz}]^T$
We are using the Newton dynamics, $x_k = x_{k-1} + v * T + a * \dfrac{-T^2}{2} - a_b * \dfrac{-T^2}{2}$

$$A_k = \begin{bmatrix} A_k^{'} & 0 & 0 \\ 0 & A_k^{'} & 0 \\ 0 & 0 & A_k^{'} \end{bmatrix}, \ A_k^{'} = \begin{bmatrix} 1 & T & \frac{-T^2}{2} \\ 0 & 1 & -T \\ 0 & 0 & 1 \end{bmatrix}$$

$$B_k = \begin{bmatrix} B_k^{'} & 0 & 0 \\ 0 & B_k^{'} & 0 \\ 0 & 0 & B_k^{'} \end{bmatrix}, \ B_k^{'} = \begin{bmatrix} \frac{-T^2}{2} & 0 & 0 \\ T & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

With the assumption that IMU readings are corrupted with Gaussian noise, we use τ_a to measure IMU noise. Similarly, τ_b is used to measure the uncertainly of the estimated acceleration bias. The process noise is designed by the piecewise white noise as described in [].

$$Q_k = \begin{bmatrix} Q_k^{'} & 0 & 0 \\ 0 & Q_k^{'} & 0 \\ 0 & 0 & Q_k^{'} \end{bmatrix}, \ Q_k^{'} = \begin{bmatrix} \frac{T^3\tau_a}{3} + \frac{T^5\tau_b}{20} & \frac{T^2\tau_a}{2} + \frac{T^4\tau_b}{8} & -\frac{T^3\tau_b}{6} \\ \frac{T^2\tau_a}{2} + \frac{T^4\tau_b}{8} & T^2\tau_a + \frac{T^3\tau_b}{3} & -\frac{T^2\tau_b}{2} \\ -\frac{T^3\tau_b}{6} & -\frac{T^2\tau_b}{2} & -T^3\tau_b \end{bmatrix}$$

$$\text{Update equation} \quad \begin{aligned} \bar{x}_k &= A_k x_{k-1} + B_{k-1} u_{k-1} \\ \bar{P}_k &= A_k P_{k-1} A_k^T + Q_{k-1} \end{aligned} \tag{3}$$

In Previous section, We seen the configuration of the system, the predicted distance between vehicle and anchor is, $\mathbf{r_k} = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2}$. This Measurement model is in nonlinear form. The correction step differs, from a linear form. Compute the measurement model **Jacobians** at x_k,

$$\mathbf{y}_k^l = \mathbf{h}(\mathbf{x}_k, \mathbf{n}_k) \qquad \mathbf{H}_k = \frac{\partial \mathbf{h}}{\partial \mathbf{x}_k}\bigg|_{\check{\mathbf{x}}_k, 0}$$

In our case mesurement step, Jacobin computed is

$$H_k = \left[\frac{x - x_i}{\bar{r}_k}, 0, 0, \frac{x - x_i}{\bar{r}_k}, 0, 0, \frac{x - x_i}{\bar{r}_k}\right]^T$$

Next steps like Kalman gain, Update Step are same as Linear one.
In practice, the IMU readings are quite noisy, which means that relying too much on the IMU is impractical as well. Usually we tend to rely more on UWB readings when tuning the covariance matrices Q and R, with the price that the jumping of UWB readings will result in sudden change of the estimated position.
To alleviate such problems, we compute the difference between predicted distance r_k and actual measurements y_k, $d_k = r_k - y_k$. If d_k is over a certain threshold, eg., 2, the update step for x_k is discarded. We also tested different models, in one of the model we added velocity update step, as our Flight controller (Pixhawk) velocity output shows better results than from this method.

3.3 Information Filter

The dual of the Kalman filter is the information filter (**IF**). standard information filter has the same assumptions, propagate through gaussian belief as the Kalman filter. The key difference is representation of gaussian belief. Gaussian are represented by (mean, covariance) in Kalman, while in information filter gaussian are in cononical form, which comprised of information matrix and an information vector. This leads to modification of the equations. The information filter tends to be more stable. The Information filter algorithm is more stable than the kalman in many robotics appilcation. When there is a large uncertainity in sensor measurement, in kalman the Σ tend to infinity, where as in information filter $\Omega = 0$. Canonical parameterization represent probability in a logarithmic form. In multi-robot problem integration of sensor data collected decentrally will be achieved by summing up through Bayes rule.

Canonical Parameterization :-
Gaussian is represented by vector(ξ) and a matrix(Ω). The Information matrix(Ω) is the inverse of covariance matrix(Σ). The Information vector(ξ) is defined as

$$\Omega = \Sigma^{-1} \qquad \Sigma - covariance$$
$$\xi = \Sigma^{-1}\,\mu \qquad \mu - mean$$

Information filter algorithm.

- $\hat{\boldsymbol{\Omega}}_k = \left(A_k \Omega_{k-1}^{-1} A_k^T + Q_k\right)^{-1}$ Prediction Step
- $\hat{\boldsymbol{\xi}} = \hat{\boldsymbol{\Omega}}_k \,\left(A_k \Omega_{k-1}^{-1} A_k^T + B_k\, u_k\right)$
- $\Omega_k = \hat{\boldsymbol{\Omega}}_k + H_k^T Q_k^{-1} H_k$ Measurement Step
- $\xi_k = \hat{\boldsymbol{\xi}}_k + H_k^T Q_k^{-1} y_k$

Extended Information Filter:
We use the same method of deriving extended kalman filter by kalman filter, here also same taylor series expansion is applied. For nonlinear measurement equation the modified equations are

$$y_k = h(\hat{\boldsymbol{\mu}}_k) + \varepsilon_k \qquad \text{nonlinear measurement step}$$
$$\xi_k = \hat{\boldsymbol{\xi}}_k + H_k^T Q_k^{-1}[y_k - h(\hat{\boldsymbol{\mu}}_k) + H_k\,\hat{\boldsymbol{\mu}}_k]$$

Information filter is the best algorithm in many robotic situations. But the Gaussian approximation of noise causes the problem in some cases. **Particle Filter** is currently one of the best estimation algorithm, which can combine measurements of any probability density distribution, but the only problem is it required more computation than these Gaussian filters.

Chapter 4

Experimental setup

We have conducted two broad experiments and validated results based on the above theories.Our full software architechture was build on ROS(Robot Operating System),which will be running on onboard computer(Raspberry pi-3). Our drone has Pixhawk Flight controller running px4 software, it provides us the acceleration and velocity data. The ground truth for all experiments was taken by the vicon (indoor positioning system based on group of infra-redcameras, which can provide millimeter level precision.)
In the **first** one, we tested four UWB sensors as anchors on the walls of the building (**Static**) as shown in Fig(3.1), We moved our vehicle fitted with tag UWB sensor.We recorded this data with ground truth as vicon.
In **Second** experiment, we tried our swarm system, in which three quadcopters are there, we fitted two UWB modules on each of the drone, as UWB modules can act as either Anchor(sending signal) or tag(reciving signal) at a time.

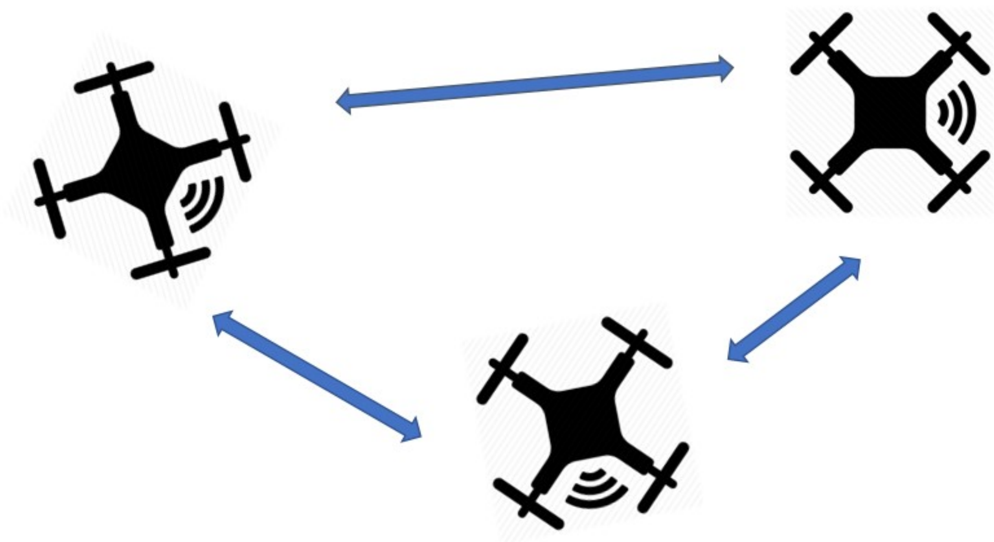
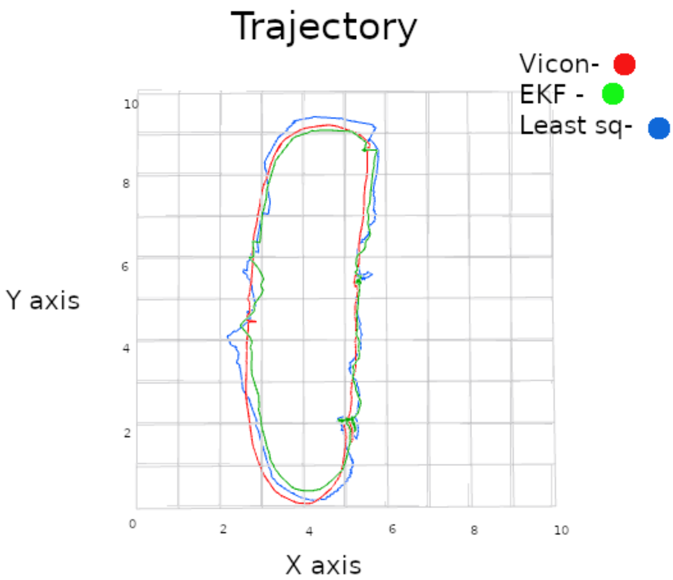
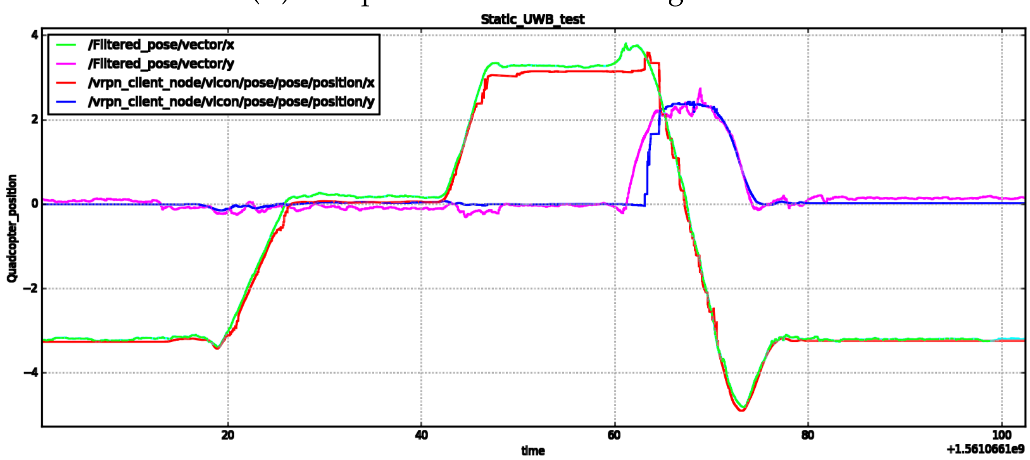


FIGURE 4.1: Setup of Working Environment

4.0.1 Results



(A) Comparison of Estimation algorithms

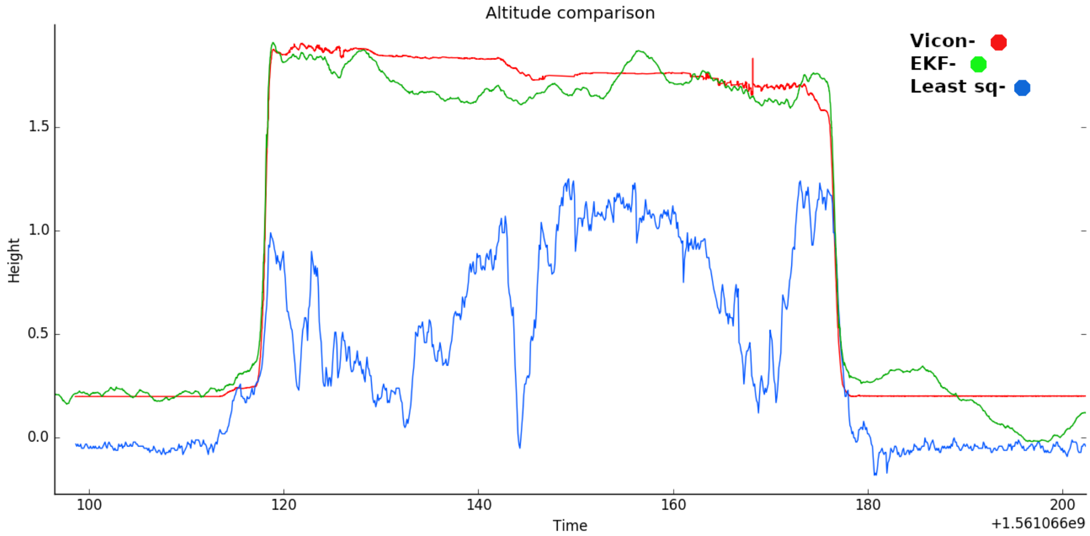


(B) varying position(x,y) with time EKF vs Vicon

FIGURE 4.2: Experiment-1 Testing of single UWB sensor on vehicle

In Experiment-1, we got a positive results, all the estimation algorithm were tested, We can see the 2D trajectory of the vehicle with ground truth as Vicon. The estimates are fair enough for precise indoor navigation, vehicle can be stable. The EKF outputs pose at 50 Hz, while least square output is at 10Hz.

We can see that the convergence of filter, we have tested this static cases many times on indoor $10m \times 10m$ area. It gave perfect results.



This is one of advantages of sensor fusion. Least square we can't do sensor usion, which led to bad estimate of height. While in EKF, we incoparated height changes due to barometer reading given by pixhawk.

Position Error	X axis	Y axis	Z axis
Least sq	0.103m	0.117m	0.653
EKF	0.096m	0.109m	0.097

TABLE 4.1: Accuracies of Experiment-1

In the second Experiment when UAV goes into unknown environemt as mentioned before the configuration is defined.

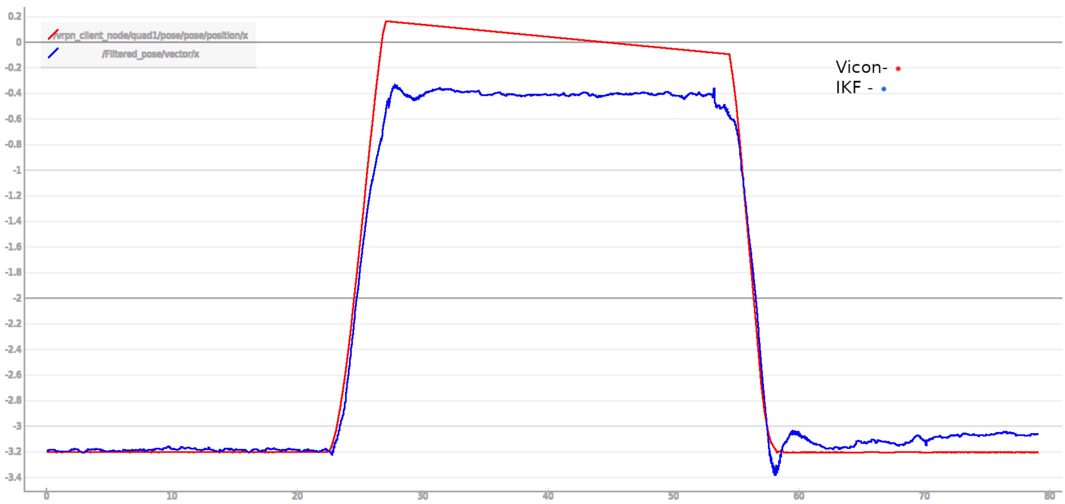


FIGURE 4.3: Experiment-2 X axis distance vs Time

The X co-ordinates that was parallel to the anchor and vehicle vector estimate was good with a error of $0.2m$. But the error in the Y co-ordinate was tremendous, it was not stable. This is one of our best estimation other estimates it diverged a lot.

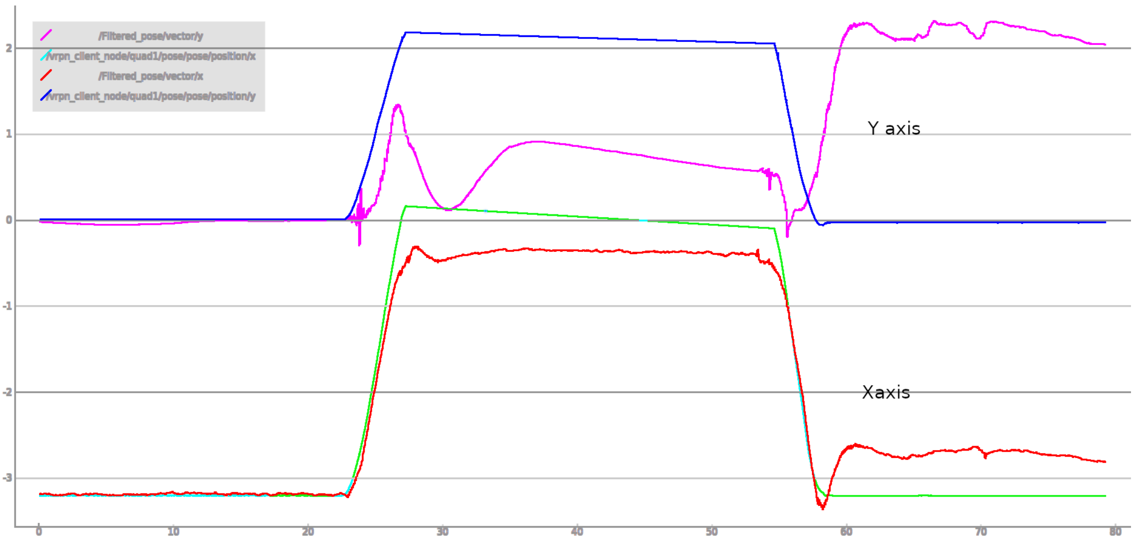


FIGURE 4.4: Experiment-2 X axis,Y axis distance vs Time

The Second Experiment result gave us answer and suggestion to many problems. We can't try Least square method in the moving anchor, as the equation cannot be reduced to a linear form $Y = A\,X$. This indoor localization method is similar to GPS, so we came to conclusion there should be atleast four anchors to determine exact location in space. Even though the exact localization, we can't obtain precisely with this method in moving anchors,we can try some other methods/filters in future. Another suggestion to this problem is with the help of relative localization if we work some methods in pathplanning we can get good result out of it.

Chapter 5

Path Planning

Motion Planninig is the process of determining where to go based on a set of objectives and goals. Based on the environment geometry, and our robot dynamcis, we want to compute a continuous sequence of collision free robot configurations connecting the initial and final configurations. There are a lot of sub-categories like Path coverage, Optimizing path, exploring unknown areas,etc., We will concentrate on Multiagent path finding.

The basic step of swarm is aggregation of robots. Self-aggregation is the grouping of certain number of objects in one place, also it is a frequent behaviour in natural world. After that we will see some navigation algorithms that will consider multiple autonomus robots and obstacles in the 3-D environment.

5.1 Boids Model

Boid Guidance Algorithms (where "Boid" stands for "Bird android") are rule-based guidance methods inspired from observations of animal flocks and swarms. It was proposed that these complex emergent behaviors could be explained if each animal agent were to follow a set of very simple rules. The combination of these rules can lead to seemingly intelligent behavior. The three simple rules are

Avoid rule is meant to prevent robots from colliding with their flockmates. Every flockmate within the avoidance range forces the boid to move away from that flockmate. Each have a region r as the safe zone, if any other comes inside, virtual force will repel both.

Align rule causes boids that are part of same group to have same general direction. The virtual force match ist heading to the average heading of the group. This will make the group to navigate in a order rather random.

Aproch makes robots move towards the center of the group of flockmates. Each robot attracted by gravitational force towards the center of all bots in neighborhood range. Bots are not scattered which will be difficult to monitor or for cooperative tasks

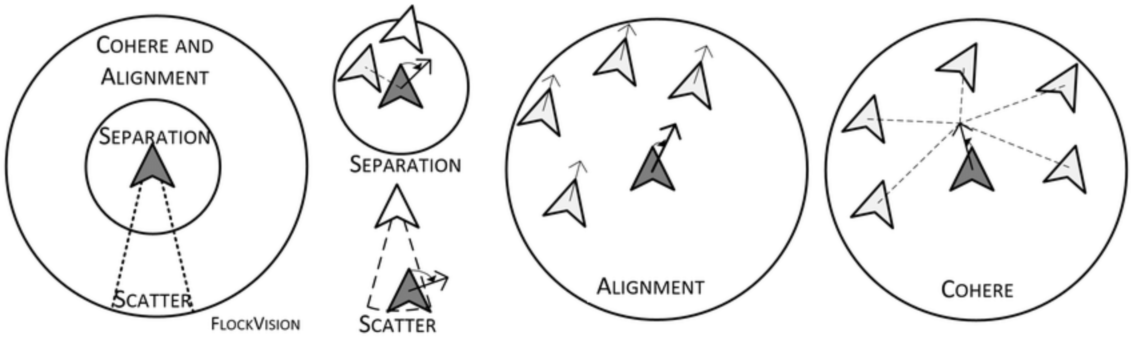


FIGURE 5.1: Three rules of Boids Algorithm

This simple algorithm can be applied in a decentralized manner to any number of vehicles. The vehicles also can navigate as a group and avoid obstacles in the path. We have implemented a basic flocking algorithm in simulation on a computer. We will try to fit our constraints of UAV's into it and test it soon.

5.2 Conflict Based Search

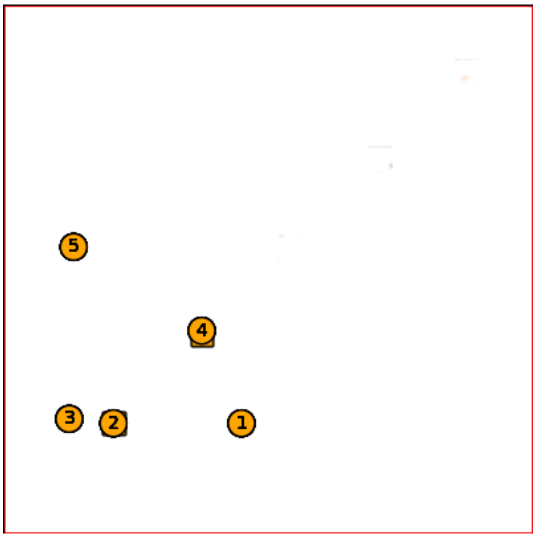
Multi-Agent Path Finding (MAPF) is ongoing research topic in both Artificial Intelligence and robotics field. MAPF solver find collision-free paths for hundreds of agents in discretized environment. Unlike flocking Algorithms, MAPF has tons of applictions like industries/warehouse where multiple drones can be used to carry objects, survey from predefined initial,final points and known map of environment. There are a family of algorithms on this topic but we specifically look on **conflict based search**

Conflict-based search uses bounded-suboptimal MAPF solver that plans for each agent independly.CBS Algorithm takes all the possible path for each and every bot from start to goal by the use of AStar.Then it recursively checks if their is any chances of collision of any bot, it rejects path having chances of collision and finally gives the path having minimum cost. Currently it was experimented by authors for 2D environment and ground bots. We can try to interpolate the method and change for the 3D environment. We will see AStar algorithm which is basic block of this algorithm.

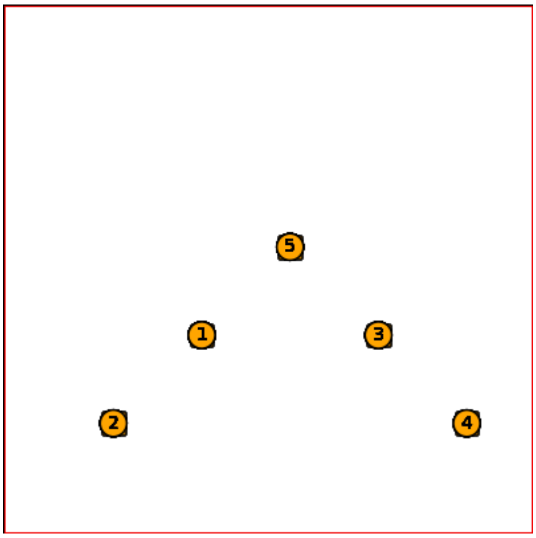
5.2.1 AStar

A* is one of the famous and optimal search algorithm in path-finding. The F, G and H are in Node class F is total cost of node. G is distance between current node and start node. H is heuristic (estimated distance from current node to end node). The main characteristics of a swarm robotics system are the following:

- **F = G + H**
- Each step expand the node with lower value of F(n).
- if goal is reached stop, else continue.



(A) initial Configuration



(B) Final Configuratioin

FIGURE 5.2: Conflict Based Search (Multi-Agent pathplanning)

Chapter 6

Conclusion

Swarm area is the main focus of research in this modern era. Many genetic algorithms are introduced to replicate the behavior of nature and apply it to robotics. In this project, we explored the main localization techniques and with their estimation algorithms. Our experiment results show that the static case of anchors performs better than the many other indoor positioning techniques. UWB solution is also very cheaper, less space, portable, less computation as compared to other vision methods. In some of the real world, examples like a light-drone show done at Singapore on an indoor auditorium is done by this method UWB. Our second experiment shows that it is difficult to estimate the accurate position with moving anchors and minimum there should be three anchors for the best position estimate. Swarm can be implemented in two ways by UWB sensors, first is fit static anchors at the wall; each vehicle has one tag sensor, giving us perfect results. Another type of swarm which can be done in an unknown environment, where we have the case of fitting anchor and tag on each vehicle (moving anchor), this method performs badly for accurate localization, but, we can perform relative localization using this method. Then we saw navigation algorithms which can be applied to both exact localization and relative localization. In the future, we will try to implement these navigational algorithms on drones and try to solve the real world problems.

Chapter 7

Reference

1. Sebastian Thrun, Wolfram Burgard, Dieter Fox " Probabilistic Robotics" MIT Press 2006.
2. Dan Simon "Optimal state estimation Kalman, H-infinity, and nonlinear approaches" Wiley-Interscience 2006.
3. KrishnaRaj,Hardik,Padey G,Kothari M. " Low Cost Solution for Pose Estimation of Quadrotor" AIAA SciTech Forum 2018.
4. Jiaxin Li, Yingcai Bi, Kun Li, Feng , M. Chen "Accurate 3D Localization for MAV Swarms by UWB and IMU Fusion"
5. Kexin Guo, Xiuxian Li, Lihua Xie. "Ultra-Wideband and Odometry-Based Cooperative Relative Localization With Application to Multi-UAV Formation Control" IEEE cybernetics
6. Chung S, Paranjape , Philip Dames, Shaojie Shen, Vijay Kumar "A Survey on Aerial Swarm Robotics". IEEE TRANSACTIONS ON ROBOTICS 2018.
7. Levent Bayındır "A review of swarm robotics tasks" ScienceDirect.
8. Ciarán McElroy, Dries Neirynck, Michael McLaughlin. "Comparison of wireless clock synchronization algorithms for indoor location systems"
9. Mohammadreza Yavari and Bradford G. Nickerson "Ultra Wideband Wireless Positioning Systems" Technical Report TR14-230
10. Wolfgang Honig, Satish Kumar, Liron Cohen, Hang Ma, Hong Xu,Nora Ayanian, Sven Koenig "Multi-Agent Path Finding with Kinematic Constraints" ICAPS 2016
11. Craig W. Reynolds Flocks, herds and schools: A distributed behavioral model
12. Hazim Shakhatreh, Ahmad Sawalmeh, Ala Al-Fuqaha, Zuochao Dou, Eyad Almaita, Issa Khalil. "Unmanned Aerial Vehicles: A Survey on Civil Applications and Key Research Challenges"