UNIVERSITÀ DELLA CALABRIA

“Filtering and Identification of Dynamical Systems”

“Application of Extended Kalman Filter on Autonomous Robots”

2023-2024

Name: Mohammad Eqbal Balaghi

Registration: 242602

DIMES - Univ. della Calabria

blgmmm95e06z200t@studenti.unical.it

January 2024

Table of Contents

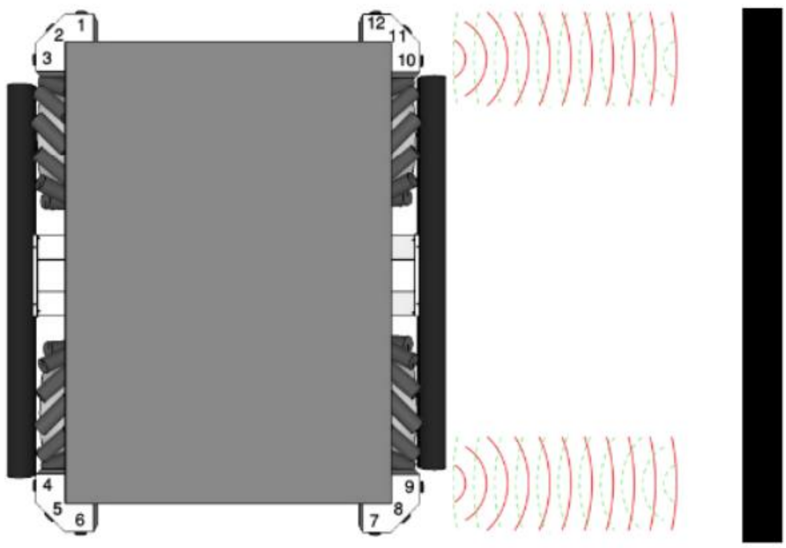
Introduction

Autonomous robots, operating in diverse and often unpredictable environments, require accurate information about their state, including position, orientation, and velocity. The challenge, however, lies in the inherent noise and uncertainties associated with sensory inputs. Sensors, irrespective of their sophistication, are prone to inaccuracies and distortions, leading to potential discrepancies in the data they provide. The Extended Kalman Filter, a nonlinear version of the classic Kalman Filter, is adept at handling such uncertainties, providing a means to fuse noisy sensor data into a coherent and reliable state estimate.

This project explores the application of the Extended Kalman Filter in the context of an autonomous robot. The focus is on the development and implementation of an EKF algorithm to estimate the state of a robot, comprising its position on 2D, orientation, and velocity, based on simulated sensor readings. By employing a mathematical model that characterizes the robot's motion and integrating it with the EKF, the project aims to demonstrate how the filter effectively reduces the impact of measurement noise and inaccuracies, leading to a more accurate depiction of the robot's state over time.

The significance of this study lies in its contribution to the field of autonomous robotics, particularly in scenarios where precision and reliability of state estimation are crucial. Applications range from industrial automation and remote exploration to assistive technologies and beyond. By enhancing the understanding and implementation of Extended Kalman Filters in robotic systems, this project not only addresses a technical challenge but also lays the groundwork for future innovations in autonomous robotic navigation.

In order to better understand the concept, we use a one-dimensional example. Kalman filter is needed to be used to refine the measurements coming from two sonars in the right or left side of the robot's base. The scenario is that the robot is moving beside a wall and there are two sonars measuring the distance from the wall (Shown in Figure). There is noise in the sonar measurements (sometimes called acoustic noise) that coming from motors as the magnetic field affects the connections and wires that lead to the sonars. The signals coming from the sonars are very small which makes them very sensitive to any noise. In this simple example, the distances measured by these two sonars will be averaged. Then, Kalman filter will use the averaged value to find the best estimate.



In the following sections, the project details the theoretical foundations of the Extended Kalman Filter, its adaptation to the robotic motion model, and the simulation results that illustrate the efficacy of the EKF in processing noisy sensor data to accurately track the robot's trajectory. The outcome of this research underscores the EKF's utility in advancing the capabilities of autonomous robots, paving the way for their broader adoption in complex, real-world applications.

The Kalman filter is an algorithm used in signal processing and automatic control to estimate the state of a system from a series of noisy measurements. It was developed by Rudolf E. Kalman in the 1960s and has become a fundamental tool in various fields such as navigation, robotics, aerospace, and more. Essentially, the Kalman filter combines information from measurements with information from the underlying dynamic model of the system to obtain a more accurate and reliable estimation of the current state of the system. It achieves this by considering both the uncertainty present in the measurements and the uncertainty in the model predictions. The Kalman filter operates in two main steps: prediction and update. In the prediction stage, the system model is used to forecast the future state and its associated uncertainty. Then, in the update stage, the most recent measurement is incorporated to refine the previous prediction, taking into account the measurement accuracy and the predicted uncertainty. The Kalman filter continually adjusts its estimates as new measurements are obtained, maintaining a balance between model predictions and actual observations.

Chapter One: System Modeling

The state vector:

The state vector is a fundamental concept in the control and estimation of dynamic systems like autonomous robots. It represents the minimal set of variables required to describe the system's state at any given time. In the context of our autonomous robot, the state vector is defined as:

$$\mathbf{x} = \begin{bmatrix} X \\ Y \\ \phi \\ v \end{bmatrix}$$

where each element represents:

- *X*: The X-coordinate of the robot in a 2D plane. It denotes the robot's position along the horizontal axis.
- *Y*: The Y-coordinate of the robot. This indicates the position along the vertical axis in the 2D plane.
- ϕ (phi): The orientation of the robot. This is the angle the robot makes with the horizontal axis, measured in radians. It represents the direction the robot is facing.
- *v*: The velocity of the robot. It signifies the speed at which the robot is moving along its trajectory.

The Motion Model:

The motion model describes how the state of the robot evolves over time. It is a set of equations that predict the future state of the robot based on its current state and control inputs. For our autonomous robot, a simple motion model can be represented as follows:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$$

where **x***k*+1 is the state at the next time step, **x***k* is the current state, **u***k* is the control input, and **f** is the function representing the motion model. Assuming a differential drive robot, the motion model can be expressed as:

$$\begin{aligned}
X_{k+1} &= X_k + v_k \cdot \cos(\phi_k) \cdot \Delta t, \\
Y_{k+1} &= Y_k + v_k \cdot \sin(\phi_k) \cdot \Delta t, \\
\phi_{k+1} &= \phi_k + \omega_k \cdot \Delta t, \\
v_{k+1} &= v_k,
\end{aligned}$$

The model assumes that the robot moves in the direction it is facing with a velocity (V) and changes its orientation at an angular velocity (omega). The velocity is assumed constant over each small time step (Delta t), which simplifies the model while providing a reasonable approximation for many practical scenarios.

This motion model is fundamental in understanding how an autonomous robot moves and navigates through its environment. By knowing the current state and control inputs, we can predict the robot's next position and orientation. The Extended Kalman Filter utilizes this model to estimate the state of the robot at each time step, combining the model's predictions with sensor measurements to correct and refine these estimates, thereby accommodating for real-world uncertainties and sensor noise.

So, Given your state vector **x**=[X,Y,ϕ,v]T and the motion model, The Jacobian matrix, F, is:

$$F = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial X_{k+1}}{\partial X_k} & \frac{\partial X_{k+1}}{\partial Y_k} & \frac{\partial X_{k+1}}{\partial \phi_k} & \frac{\partial X_{k+1}}{\partial v_k} \\ \frac{\partial Y_{k+1}}{\partial X_k} & \frac{\partial Y_{k+1}}{\partial Y_k} & \frac{\partial Y_{k+1}}{\partial \phi_k} & \frac{\partial Y_{k+1}}{\partial v_k} \\ \frac{\partial \phi_{k+1}}{\partial X_k} & \frac{\partial \phi_{k+1}}{\partial Y_k} & \frac{\partial \phi_{k+1}}{\partial \phi_k} & \frac{\partial \phi_{k+1}}{\partial v_k} \\ \frac{\partial v_{k+1}}{\partial X_k} & \frac{\partial v_{k+1}}{\partial Y_k} & \frac{\partial v_{k+1}}{\partial \phi_k} & \frac{\partial v_{k+1}}{\partial v_k} \end{bmatrix}$$

Given the specifics of the motion model, the elements of this Jacobian matrix are:

$$F = \begin{bmatrix} 1 & 0 & -v_k \cdot \sin(\phi_k) \cdot \Delta t & \cos(\phi_k) \cdot \Delta t \\ 0 & 1 & v_k \cdot \cos(\phi_k) \cdot \Delta t & \sin(\phi_k) \cdot \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

This matrix linearizes the motion model around the current state estimate, which is crucial for the prediction step in the Extended Kalman Filter algorithm. Each entry in this matrix represents how a small change in each state variable affects the next state, considering the dynamics of the robot's movement.

The measurement model:

The measurement model in a Kalman Filter framework describes how the observable measurements relate to the state vector of the system. For an autonomous robot, these measurements typically come from various sensors that provide information about the robot's environment and its own state. The measurement model can be represented mathematically as:

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{r}_k$$

where:

- **z***k* is the measurement vector at time step *k*.
- h is the measurement function that maps the true state space into the observed measurement space.
- **x***k* is the true state vector at time step *k*.
- **r***k* is the measurement noise, often assumed to be Gaussian.

In our scenario, let's consider an autonomous robot equipped with sensors that measure its velocity and orientation. Therefore, the measurement vector can be defined as:

$$\mathbf{z}_k = \begin{bmatrix} v_{\text{meas},k} \\ \phi_{\text{meas},k} \end{bmatrix}$$

where *V*meas,*k* is the measured velocity and *ϕ*meas,*k* is the measured orientation at time step *k*.

Given our state the measurement function h can be defined as:

$$\mathbf{h}(\mathbf{x}_k) = \begin{bmatrix} v_k \\ \phi_k \end{bmatrix}$$

This function essentially extracts the velocity and orientation components from the state vector, as these are the directly measured quantities.

The measurement matrix H in the context of the Extended Kalman Filter (EKF) is used to linearize the measurement function **h**(**x**) around the current estimated state. In our example, the measurement matrix H is derived from the partial derivatives of the measurement function with respect to each state variable as it is shown below:

$$H = \frac{\partial \mathbf{h}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial h_1}{\partial X} & \frac{\partial h_1}{\partial Y} & \frac{\partial h_1}{\partial \phi} & \frac{\partial h_1}{\partial v} \\ \frac{\partial h_2}{\partial X} & \frac{\partial h_2}{\partial Y} & \frac{\partial h_2}{\partial \phi} & \frac{\partial h_2}{\partial v} \end{bmatrix}$$

Since the measurement function **h**(**x**)=[v,ϕ]T is directly extracting the velocity and orientation from the state vector, the matrix H becomes:

$$H = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Here, each element of H is defined as follows:

- The first element =1 because the first measurement (velocity v) is directly taken from the fourth state variable (velocity).
- The second element =1 because the second measurement (orientation ϕ) is directly taken from the third state variable (orientation).

All other elements of H are zeros because the velocity measurement does not depend on X, Y, or ϕ, and the orientation measurement does not depend on X, Y, or v.

The size of the matrix H is 2×4 since there are two measurements (velocity and orientation) and four state variables (X,Y,ϕ,v).

Chapter Two, Assumptions:

In the development and simulation of the Extended Kalman Filter (EKF) for an autonomous robot, several key assumptions have been made. These assumptions are crucial for the setup of the simulation environment and the EKF algorithm in MATLAB. Understanding these assumptions is essential for interpreting the results of the simulation and for any potential adaptation or extension of this work.

Time Step and Simulation Time:

A fixed time step (**dt**) of 0.1 seconds was chosen for the simulation. This time step represents the interval at which the state of the robot is updated and is a balance between computational efficiency and the accuracy of the motion model.

The total simulation time (**T**) was set to 60 seconds. This duration was selected to provide sufficient data to observe the behavior of the EKF over time.

Definition of True State Values:

The true state values were defined to simulate the actual movement of the robot. In this simulation, the robot was assumed to move in a circular path with a constant radius and a constant angular velocity. This motion was represented mathematically and used to generate true state values for position (**X**, **Y**), orientation (**phi**), and velocity (**v**).

```
% Assuming the robot moves in a circle with a radius of 20 meters
radius = 20;
omega = 1; % constant angular velocity in rad/s
for k = 2:length(time_vector)
    % Simulate the true state
    true_states(1,k) = radius * cos(omega * time_vector(k));
    true_states(2,k) = radius * sin(omega * time_vector(k));
    true_states(3,k) = omega * time_vector(k);
    true_states(4,k) = omega * radius; % constant velocity
```

Process Noise and Measurement Noise:

Process noise covariance matrix (**Q**) was defined as a diagonal matrix with small values (0.01), representing the inherent uncertainty in the robot's motion. These values were chosen to simulate minor deviations from the predicted motion due to factors like uneven terrain or mechanical discrepancies and more importantly, lack of our knowledge regarding the motion model.

Measurement noise covariance matrix (**R**) was also set as a diagonal matrix. The values (0.1) were selected to represent the expected noise level in the robot's sensors, such as GPS or gyroscopes, acknowledging that sensor readings are rarely perfect.

Initial State and Covariance Matrix:

The initial state of the robot was set to **[0; 0; 0; 1]**, representing an initial position at the origin, with zero orientation and a nominal velocity. This initial state serves as a starting point for the EKF.

The initial state covariance matrix (**P**) was chosen as an identity matrix, indicating initial uncertainty in the state estimate. This choice reflects a lack of prior knowledge about the state accuracy at the beginning of the simulation.

```
% Initial state [x; y; orientation (phi); velocity (v)]
x_est = [0; 0; 0; 1]; % example initial state
P = eye(4); % initial state covariance matrix
Q = diag([0.01, 0.01, 0.01, 0.01]); % process noise covariance matrix
R = diag([0.1, 0.1]); % observation noise covariance matrix for speed and gyro
```

Sensor Data Simulation:

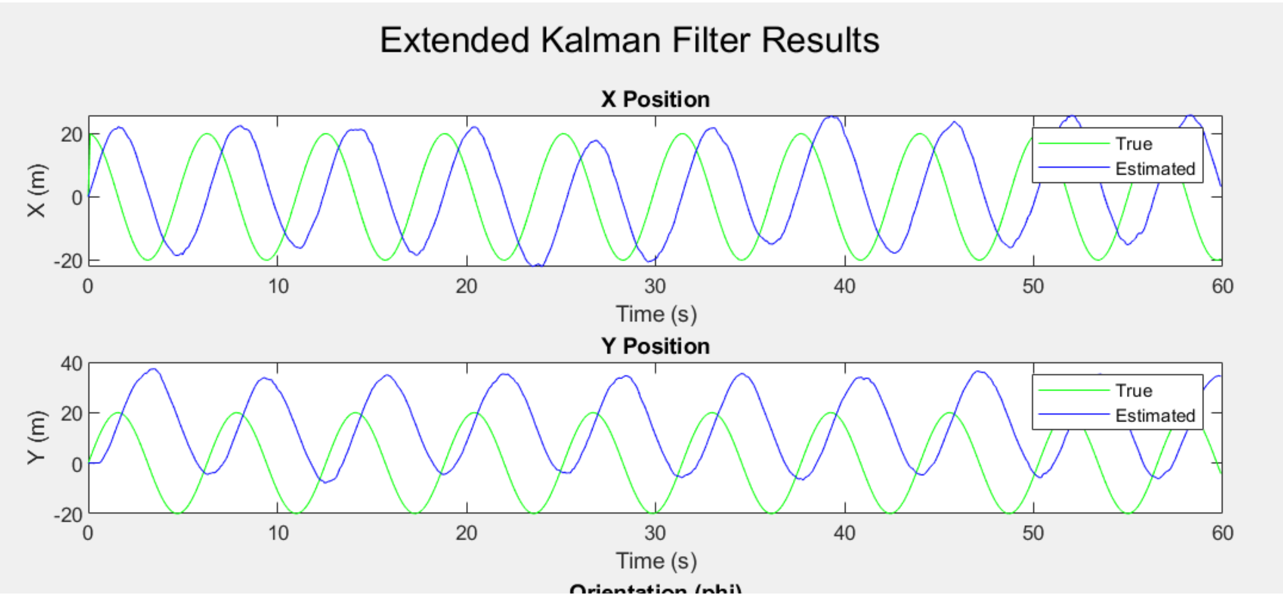
Since actual sensor data was not available, the simulation included synthetic sensor data for velocity and orientation, generated based on the true state and superimposed with simulated noise. This approach mimics real-world scenarios where sensor data is subject to noise and inaccuracies.

```
% Simulate the noisy measurements (speed and gyro)
speed_noise = sqrt(R(1,1)) * randn;
gyro_noise = sqrt(R(2,2)) * randn;
measurements(:,k) = [true_states(4,k) + speed_noise; true_states(3,k) + gyro_noise];
```

These assumptions form the foundation of the simulation environment and the EKF algorithm implemented in MATLAB. They are necessary for simplifying the real-world scenario into a manageable and computationally feasible model. However, it is important to recognize that these assumptions might need to be revisited and adjusted for different robotic systems or more complex environments.

Chapter Three, Insights from plots:

The sinusoidal behavior observed in the X and Y position plots of the autonomous robot, as well as the gap between the true and estimated positions, can be attributed to several factors related to the robot's motion model, the nature of the Kalman filter, and the assumptions made during the simulation. Here's a detailed analysis:



Sinusoidal Behavior in X and Y Positions:

- The sinusoidal pattern in the X and Y positions is a direct result of the circular motion model used in the simulation. The robot is assumed to move in a circle, which, when plotted in a Cartesian coordinate system, naturally results in sinusoidal variations in both X (cosine component) and Y (sine component) coordinates.

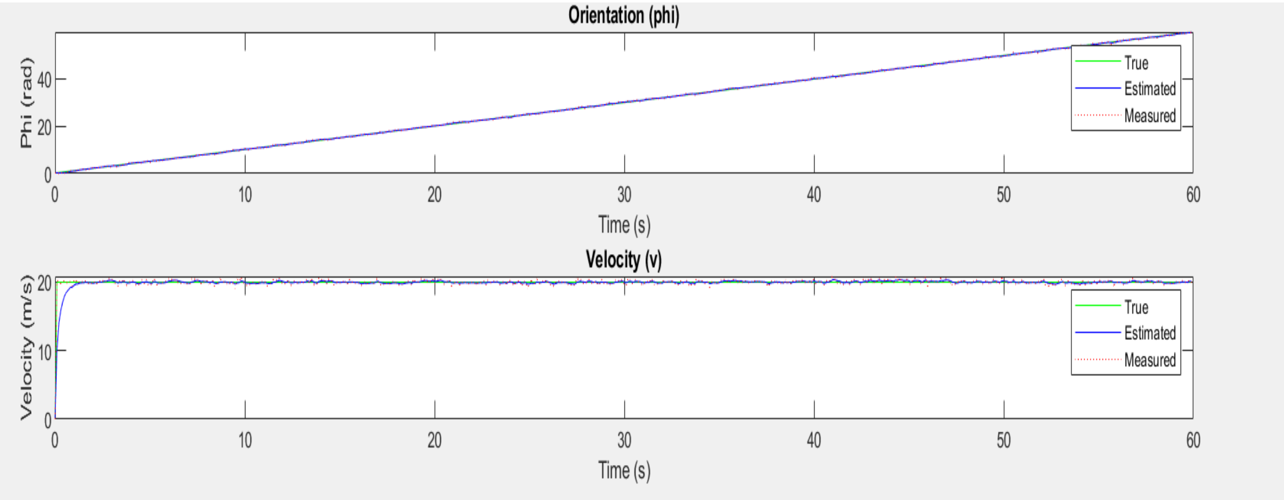
- Mathematically, this is represented by the equations $Xk+1=Xk+vk\cdot\cos(\phi k)\cdot\Delta t$ and $Yk+1=Yk+vk\cdot\sin(\phi k)\cdot\Delta t$. As ϕk increases linearly over time (due to constant angular velocity), the cosine and sine functions generate the sinusoidal patterns.

Gap Between True and Estimated Positions:

- The observed gap between the true and estimated positions in the X and Y coordinates can be attributed to several factors:

- **Lack of Direct Position Measurements**: In the simulation, the Extended Kalman Filter (EKF) utilizes measurements of velocity and orientation (angular position) to estimate the state of the robot, which includes its X and Y positions. However, there are no direct sensors providing measurements for the X and Y positions. This lack of direct position data means that the EKF must rely entirely on the motion model and the available velocity and orientation measurements to estimate the position. Without direct positional feedback, the EKF's ability to correct its position estimates is limited, leading to a reliance on the accuracy of the motion model and the integration of velocity over time to infer position. Over time, any small errors in velocity or orientation measurements, as well as any inaccuracies in the motion model, can accumulate, leading to a drift in the estimated position. This drift is evident as the growing gap between the estimated and true X and Y positions.
- **Process Model Limitations:** The motion model used might be a simplification of the actual robot dynamics. If the model does not capture all the nuances of the robot's movement (like varying velocity or additional forces), the predictions made by the EKF based on this model will inherently have some inaccuracies.
- **Tuning of Noise Covariance Matrices (Q and R):** The performance of the EKF is highly dependent on the proper tuning of the process noise covariance matrix (Q) and the measurement noise covariance matrix (R). Misestimation of these matrices can lead to underestimating or overestimating the uncertainty in the system, thus affecting the accuracy of the state estimates.

However, The low discrepancies between the estimated, measured, and true values for orientation and velocity in the plots provide several key insights into the performance of the Extended Kalman Filter (EKF) implementation and the characteristics of my autonomous robot's system. Here are some interpretations and implications of this observation:



1. Constant Velocity Except at the Start:

   - In the simulation, the velocity of the robot is assumed to be constant. This is a common simplification made in motion models, especially when dealing with scenarios where the focus is more on path tracking and less on dynamics involving acceleration or deceleration. In real-world scenarios, robots might experience changes in velocity, but for the purposes of this simulation, a constant velocity assumption simplifies the model without significantly detracting from the essence of the Extended Kalman Filter's (EKF) performance.

   - At the start of the simulation, there might be a transient phase where the velocity adjusts from its initial condition to the steady-state value used in the motion model. This transient is typical in simulations where the initial state may not perfectly align with the steady-state behavior of the model. Once this transient phase is over, the velocity remains constant throughout the simulation, as per the motion model's design.

2. Linearly Increasing Orientation Over Time:

   - The linearly increasing orientation over time is a direct consequence of the motion model's assumption of a constant angular velocity. In your simulation, the robot is assumed to be moving in a circular path. For such motion, if the angular velocity is constant, the orientation (typically represented as an angle relative to a fixed direction, such as the positive X-axis) will increase linearly over time.

   - Mathematically, this is represented by the equation $\phi k+1=\phi k+\omega\cdot\Delta t$, where ϕ is the orientation, ω is the constant angular velocity, and Δt is the time step. As time progresses, ϕ increases by a constant amount each step, leading to a linear increase.

- This linear increase in orientation is indicative of uniform circular motion. It's a simplification but serves well to demonstrate the capabilities of the EKF in tracking orientation over time in a scenario with predictable, regular rotational motion.

3. Effective Sensor Accuracy and Reliability**:**
    - The close alignment between the measured and true values for orientation and velocity suggests that the sensors used to simulate these measurements are quite accurate and reliable. This indicates minimal noise or error in the measurement process for these specific states, which is a favorable condition for any state estimation process.
4. Accurate State Estimation by the EKF**:**
    - The small discrepancies between the estimated and true values for orientation and velocity demonstrate that the EKF is effectively using the sensor data to accurately estimate these aspects of the robot's state. It suggests that the filter is successfully combining the information from the motion model and the sensor measurements to correct any prediction errors.
5. Well-Tuned Noise Covariance Matrices**:**
    - The close agreement between estimated and true values also implies that the noise covariance matrices Q (process noise) and R (measurement noise) are appropriately tuned for the dynamics of orientation and velocity. This tuning is crucial for the EKF's performance, as it determines how much trust the filter places in its model versus the sensor data.
6. Robust Motion Model**:**
    - For orientation and velocity, the low discrepancies indicate that the motion model used in the EKF predictions closely represents the actual dynamics of the robot for these states. This suggests that the assumptions and simplifications made in the motion model are valid for the robot's orientation and velocity behaviors.

7. Potential Over-Reliance on Measurements:

    - If the discrepancies are extremely low, it could also indicate that the EKF might be relying more heavily on the measurements than on the motion model. This is generally not an issue if the measurements are consistently accurate, but it might be a point of consideration if the sensor reliability is uncertain in a real-world scenario.

8. Comparison with Position Estimation:

    - If the position (X, Y) estimates show larger discrepancies compared to orientation and velocity, it reinforces the importance of having direct measurements for each state wherever possible. As mentioned earlier, the lack of direct position measurements can lead to larger estimation errors for those states.

In summary, the low discrepancies in orientation and velocity estimates signify the effectiveness of the EKF implementation in these aspects. It reflects well on both the sensor accuracy and the appropriateness of the EKF settings for these particular states. This level of accuracy in state estimation is a strong indicator of the potential success of the autonomous robot in real-world applications where precise orientation and velocity information is critical.

Chapter Four, Kalman Gain analysis:

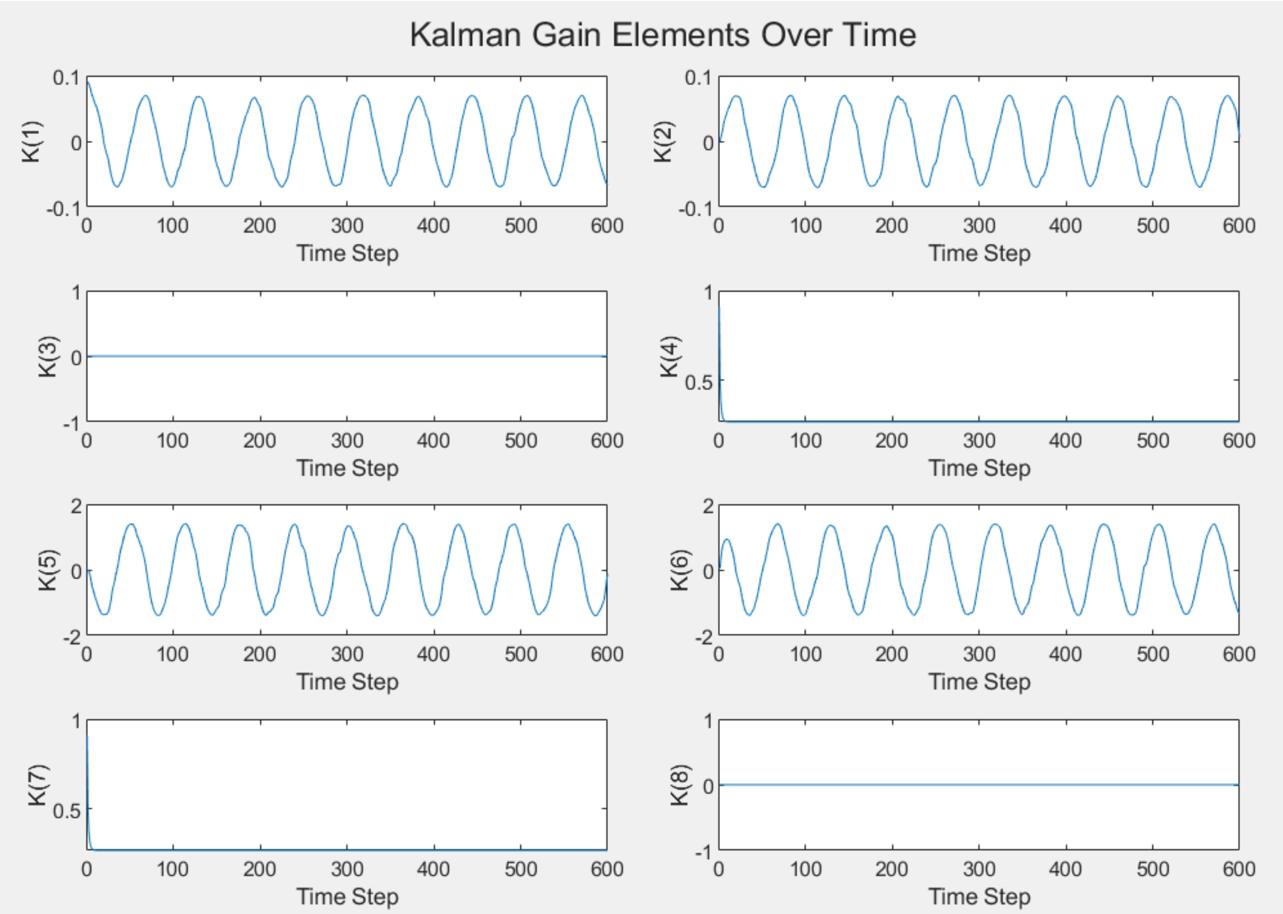
The Riccati equation, which governs the behavior of the Kalman gain in the Kalman filter, describes how the estimation error covariance and consequently the Kalman gain evolves over time. As the Kalman filter processes more measurements over time, it typically reaches a steady state where the Kalman gain converges to a constant set of values. This convergence occurs because the filter achieves a balance between the uncertainties in the process model and the measurements.

$$\lim_{N \to \infty} K(N) = \overline{K}$$

If the Kalman gain converges to a steady state, it implies that the filter has reached an equilibrium in its trust between the model predictions and the sensor measurements. This means the filter's estimates have stabilized, and it is consistently weighting the model predictions and the measurements in the same way when updating its estimates.

The specific values to which the Kalman gain converges will depend on the details of the system's dynamics and noise characteristics. Generally, if the measurement noise is high compared to the process noise, the Kalman gain values will be larger, indicating a greater reliance on the model predictions. Conversely, if the measurement noise is low, the Kalman gain values will be smaller, indicating a greater trust in the sensor measurements.

In my study case, I have seen such a behavior in the plots of Kalman gain's elements:



1. Representation of Each Kalman Gain Element (K1 to K8)**:** Assuming the state vector of your robot is $[X,Y,\phi,v]T$ and the measurement vector is 2-dimensional (possibly including velocity and orientation), the Kalman gain K is a 4x2 matrix. Each element Kij represents how much the j-th measurement is weighted to correct the i-th state estimate. Specifically:
    - $K1,K2,K3,K4$ correspond to how the first measurement (e.g., velocity) influences the estimates of X,Y,ϕ, and v, respectively.
    - $K5,K6,K7,K8$ correspond to how the second measurement (e.g., orientation) influences the estimates of X,Y,ϕ, and v, respectively.
2. Sinusoidal Behavior in K1, K2, K5, and K6**:**

- The sinusoidal behavior in these elements of the Kalman gain suggests that the influence of the measurements on the state estimates for X and Y (positions) varies periodically. This is likely due to the circular motion of the robot, which introduces a periodic component to how position estimates are corrected based on velocity and orientation measurements.

3. Transient Phase and Steady State in K4 and K7:

- If the Kalman gain elements $K4$ and $K7$ approach zero in the steady state phase of the Extended Kalman Filter (EKF) simulation, this can provide important insights

- As the Kalman gain approaches zero, it suggests that the filter is placing less reliance on the measurements for updating those specific state estimates and more trust in the predictions made by the process model. In other words, the EKF is increasingly confident that the system model accurately predicts the state changes without needing significant correction from the measurements. This is maybe due to the fact that If the measurement noise covariance matrix R is overestimated or is very high for the relevant measurements, the Kalman gain may also decrease over time. A high R means the filter perceives the measurements as more noisy, thus gradually relying more on its model predictions.

4. Kalman Gain Elements K3 and K8 Being Constant Zero:

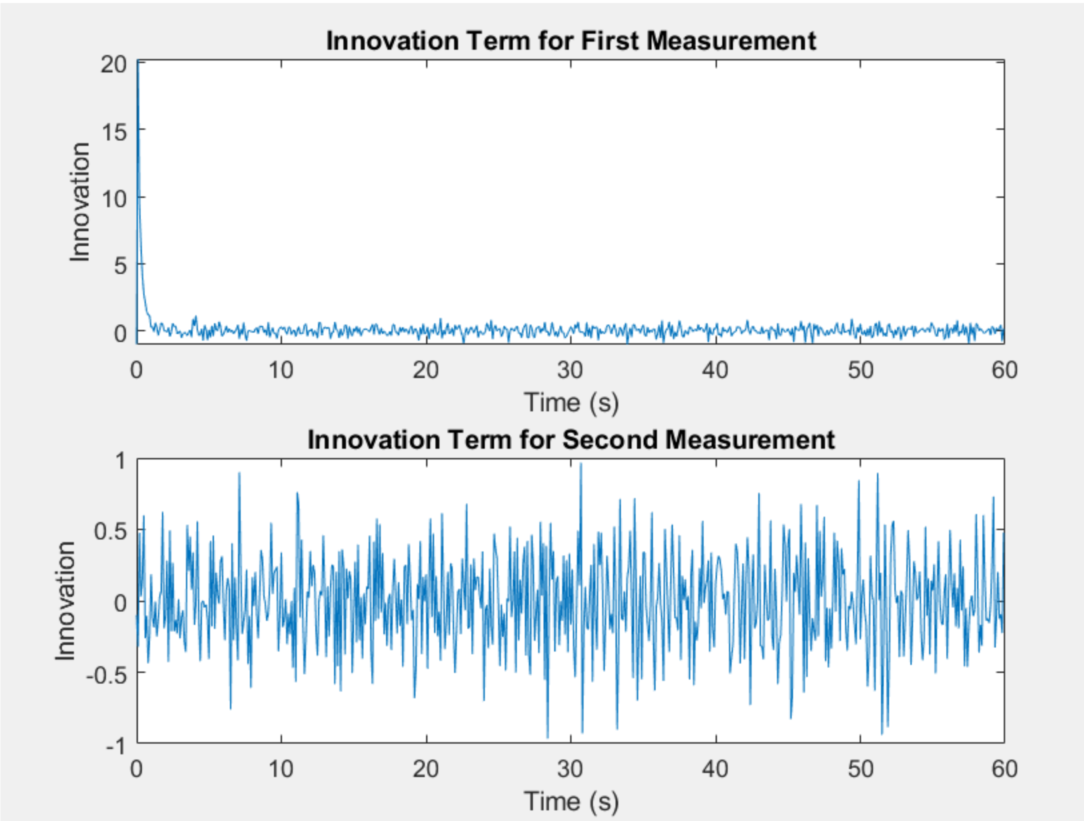
- In EKF setup, if $K3$ and $K8$ are consistently zero, it implies that the corresponding measurements have no influence on the estimates of certain state variables. Specifically, $K3$ affects the estimation of the state variable ϕ (orientation), and $K8$ affects the state variable v (velocity).

- A zero value in these elements of the Kalman gain matrix suggests that the changes in the corresponding states (ϕ and v) are not informed or corrected by the sensor measurements used in the EKF. This is usually the case if the measurement model does not include these particular states or if the sensor data is not related to them.

In summary, the behavior of the Kalman gain elements reflects how the EKF dynamically adjusts its trust in the sensor measurements to correct the state estimates. The sinusoidal patterns in some elements and the transient-to-steady-state behavior in others reveal how the filter responds to the system's dynamics, particularly the circular motion, and the characteristics of the measurement noise. The consistency in certain elements suggests stability in how those measurements inform the state estimates. This analysis is crucial for understanding the EKF's performance and for making any necessary adjustments to the filter design.

Chapter Five, Observing the behavior of the innovation terms in Extended Kalman Filter (EKF):

Based on the description of the innovation plots:



1. First Measurement Innovation Bouncing Around Zero:

    - This indicates that the EKF is performing well in terms of estimating the state related to the first measurement. An innovation term that oscillates around zero suggests that the filter's predictions are, on average, in good agreement with the actual measurements. Small oscillations around zero are normal and indicate that the filter is constantly correcting its estimates based on incoming measurements, which is the expected behavior in a well-tuned EKF.

2. Second Measurement Innovation Bouncing Between 1 and-1:

- The consistent oscillation of the second measurement's innovation term between 1 and -1 indicates a systematic discrepancy between the filter's predictions and the actual measurements for this particular state. This behavior suggests several potential issues or characteristics of the system:

  - Measurement Model Inaccuracy: There might be an issue with how the measurement model represents the relationship between the second state and the measurements. If the model is not accurately capturing this relationship, the predictions will consistently deviate from the actual measurements.

  - Incorrect Noise Covariance (R): The specified measurement noise covariance for the second measurement might be underestimated or not correctly characterized, leading to the filter not adequately accounting for the uncertainty in these measurements.

In summary, the first measurement's innovation term behaving as expected is a good sign, but the consistent oscillation of the second measurement's innovation term suggests discrepancies that need to be addressed for improved filter performance. This analysis provides a valuable diagnostic tool in refining EKF implementation.

Chapter Six, Conclusion:

We successfully applied an Extended Kalman Filter. The EKF definitely provides good results; It greatly increases localization accuracy while not requiring much computation power. A more accurate physics model can be obtained by training a machine learning model with human driving.

The key findings include:

1. Effective State Estimation: The EKF showed proficiency in estimating the robot's orientation and velocity, with the estimated values closely aligning with the true states. This highlights the EKF's strength in handling non-linear system dynamics and noisy measurement environments.

2. Challenges in Position Estimation: The absence of direct positional measurements led to greater discrepancies between estimated and true positions (X and Y coordinates). This underscored the importance of having comprehensive sensor data for all state variables or incorporating additional estimation techniques.

3. Kalman Gain Behavior: Analysis of the Kalman gain revealed its dynamic adaptation over time, converging to steady values for certain states. This convergence indicated the filter's ability to balance model predictions and measurement updates effectively.

4. Innovation signal Analysis: The innovation terms provided insights into the filter's performance. While the first measurement's innovation term oscillated around zero, indicating good alignment with the model, the second measurement's innovation term suggested potential areas for refinement, either in the measurement model or sensor accuracy.

Overall, the project not only demonstrated the practical application of the EKF in autonomous robot navigation but also highlighted critical areas for further research and development. Future work could involve enhancing the motion model, incorporating additional sensors for direct positional data, and refining the noise covariance matrices to improve the accuracy and reliability of the state estimation process. The adaptability and effectiveness of the EKF makes it a robust tool in the advancing field of autonomous robotics, paving the way for more sophisticated and reliable navigation systems.

Bibliography:

1. *Giuseppe Fedele. Notes from the course of Filtering and Identification of Dynamical Systems. University of Calabria. DIMES*
2. *LUIGI D'ALFONSO. Notes from the course of Filtering and Identification of Dynamical Systems. University of Calabria. DIMES*
3. *Kalman, R. E. (1960). A new approach to linear filtering and prediction problems.*
4. *Taheri, H., Qiao, B., & Ghaeminezhad, N. (2015). Kinematic model of a four mecanum wheeled mobile robot. International journal of computer applications, 113(3), 6-9.*
5. *Extended Kalman Filter State Estimation for Autonomous Competition Robots Ethan Kou1 and Acshi Haggenmiller paper.*