The University of Texas at Austin
Aerospace Engineering
and Engineering Mechanics
Cockrell School of Engineering

March 2023

DESIGN REVIEW #3

USRC - Simulation / Estimation

Nick Franken
Izaac Facundo
Jeremy Morrison
Neel Pandey
Preston Thomas
William Wang

PROJECT OVERVIEW

OUR PROJECT

- The overall need of this project is to develop drones capable of sensing and avoiding obstacles with a minimized risk of collision or damage to property in GPS-denied environments.
  - Construct a system that is able to take real time inputs from actively flying drones and output a state estimation with 3-D probability ellipsoid visualization using an Extended Kalman Filter (EKF)
  - Avoid collisions with other drones and environmental obstacles
- The application of this system would be for the increase of commercial drone delivery operations

ESTIMATION & SIMULATION

TEAM STRUCTURE

- Meeting Times:
  - Wednesday (Lecture): 11AM-12PM
  - Wednesday (Lab) 3-6PM
  - More time if needed
- Informal Split of responsibilities
  - Nick is person of contact
  - Expertise distributed among us

GOALS

- Implement an extended Kalman filter (EKF) on drone positions (done)
- Implement flight control software to enable basic flight capabilities (ongoing)
- Visualize drones and drone behavior using simulation software (done)
- Develop drones' ability to successfully avoid obstacles
- Develop architecture for communication between drones
- Design system for software-software and software-hardware integration

PROFESSIONAL RESPONSIBILITY

- Collision Avoidance: Need a nearly guaranteed success rate of avoiding obstacles, inanimate and animate alike, to minimize injury and/or damage to property
- This affects certain aspects of our drone operation:
  - Maximum velocity
  - Altitude
  - Avoidance trajectories
- More factors will be considered in the future
  - Acoustics
  - Drone end of life

PROJECT UPDATES

PROJECT UPDATES

- 3D Probability Ellipsoid
  - Adapted Potential Flow Field simulation to drone simulation with physical dynamics/sensor models
- Sensor Investigation
  - Researching sensor integration with simulation
  - Researching implementation of sensors onto physical drone
- PX4/QGroundControl Integration
  - Working with communications to complete flashing of PX4 onto Pixhawk
  - Research integration of PX4 with simulation

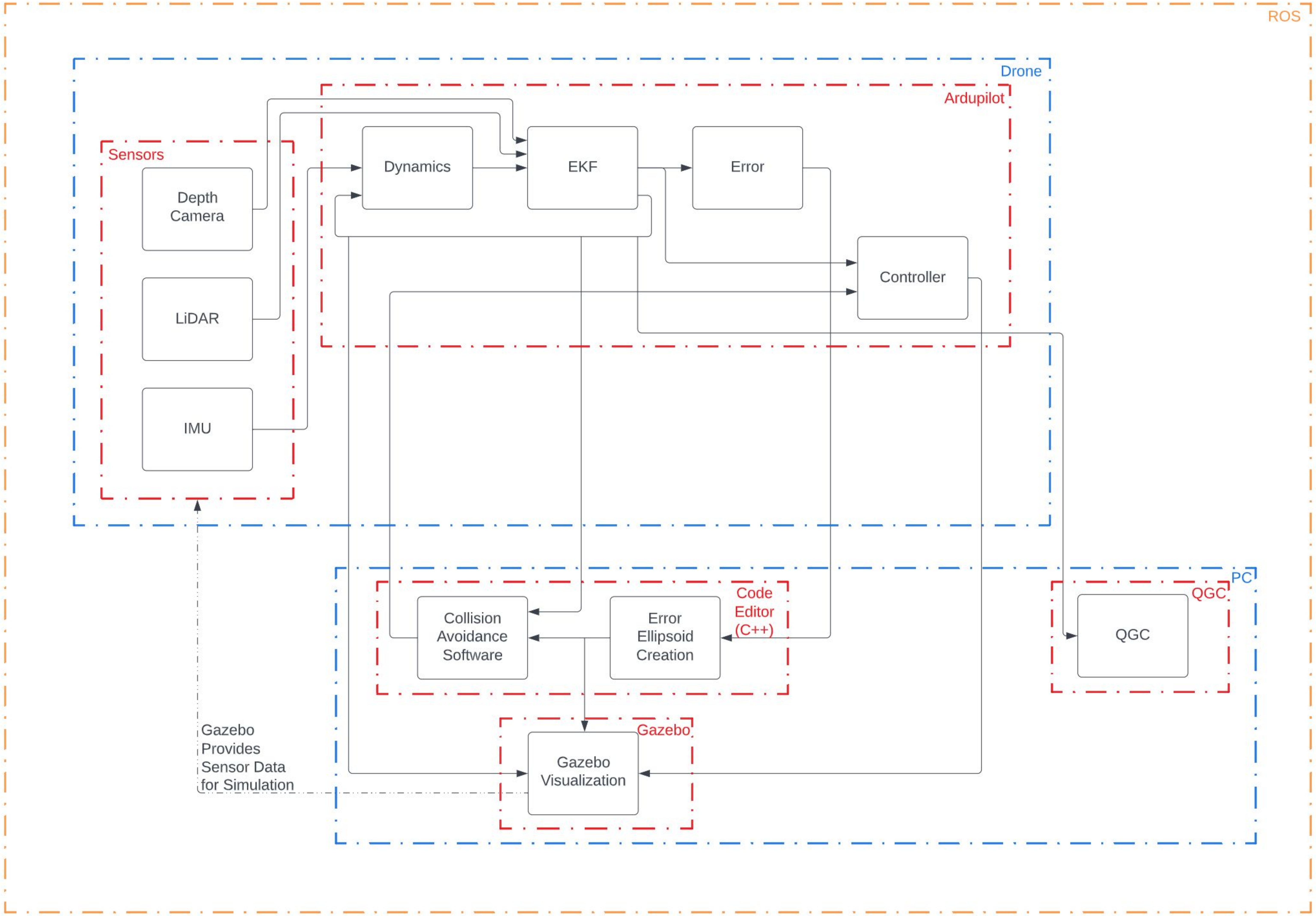
SOFTWARE & BLOCK DIAGRAM

OVERALL OPERATING SYSTEM (OS)

- For developmental purposes, we will be using Linux/Ubuntu. This will allow for more freedom and control when developing software. Certain software like the Robot Operating System (ROS) are developed specifically for operation on a Linux OS.

SOFTWARE LIST

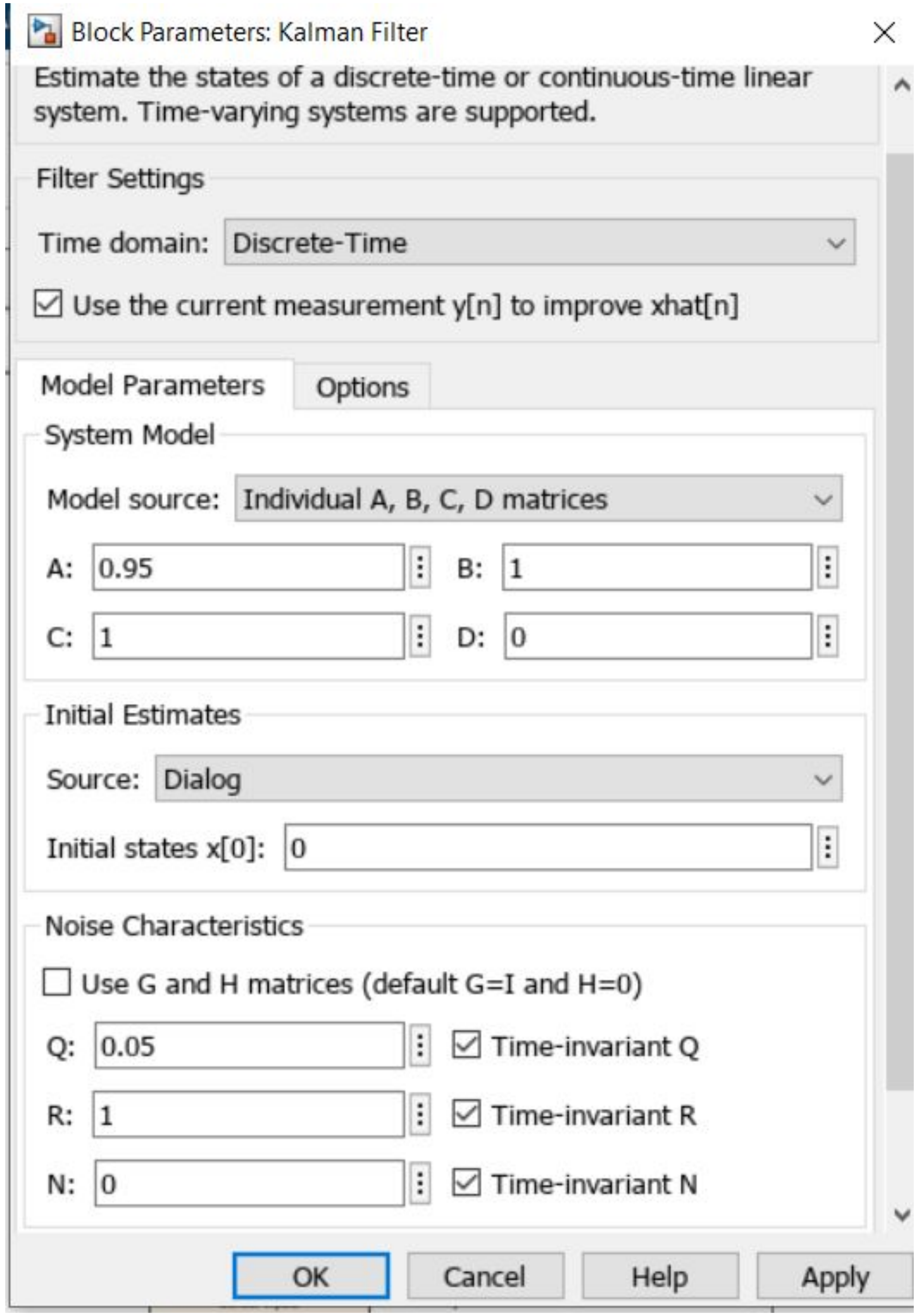
- PX4
- Code Editor (MATLAB, VSCode, Visual Studio, Eclipse, etc.)
- Gazebo/Simulink
- QGroundControl (QGC)
- Robot Operating System (ROS) / MAVLink (“MAVROS”)
- Sensor Software (LiDAR, Depth Camera, etc.)

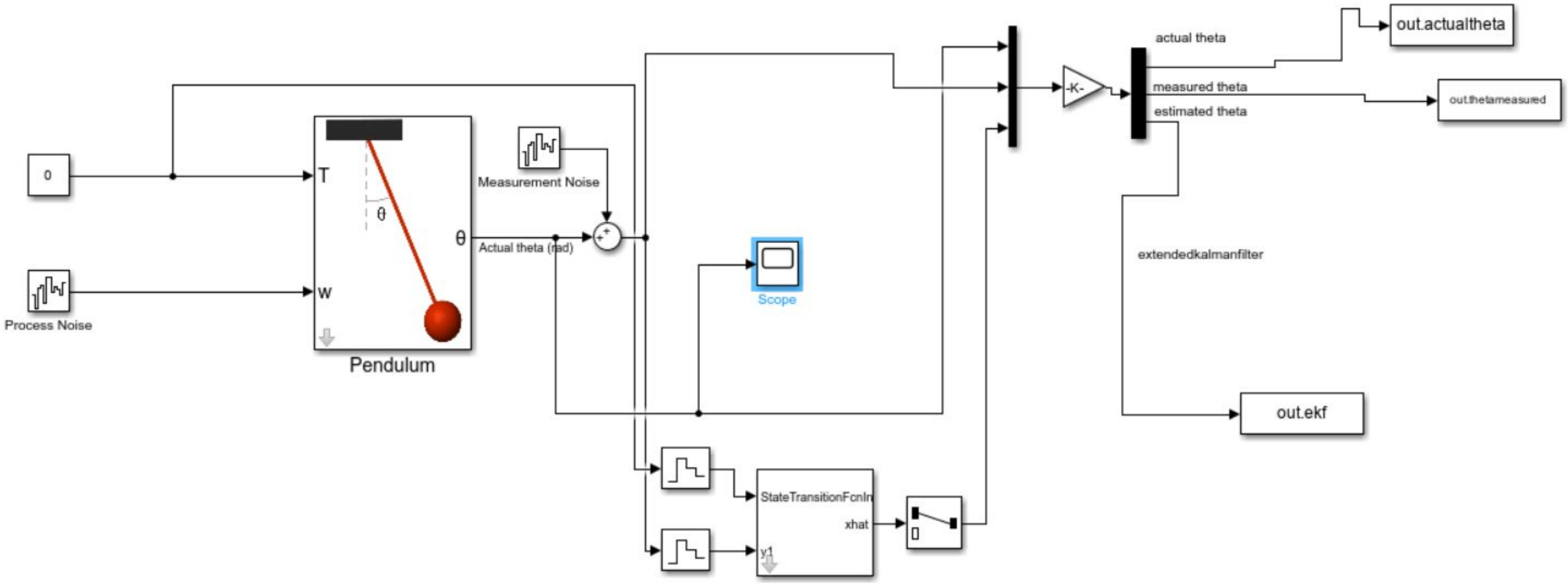

ROS
Drone
Ardupilot
Sensors
Depth Camera
LiDAR
IMU
Dynamics
EKF
Error
Controller
PC
Code Editor (C++)
Collision Avoidance Software
Error Ellipsoid Creation
QGC
QGC
Gazebo
Gazebo Visualization
Gazebo Provides Sensor Data for Simulation

PENDULUM SIMULATION

Extended Kalman Filter (EKF)

- A Kalman Filter is a filter that takes a less than perfect dynamic model and noisy measurements to provide very accurate state estimations for a system. This filter only works for linear systems.
- Two covariance matrices: Q and R (process and measurement noise respectively)
- An EKF can be used in non-linear systems but requires more computation. The general concept is the same, and the Q and R covariance matrices are still very important.




0
Process Noise
T
w
θ
θ
Pendulum
Measurement Noise
Actual theta (rad)
Scope
StateTransitionFcnIn
xhat
y1
-K-
actual theta
measured theta
estimated theta
out.actualtheta
out.thetameasured
extendedkalmanfilter
out.ekf

POTENTIAL FIELD 3D SIMULATION

Potential Field Dynamics:

- The main motivation in creating a potential field model was to have a three dimensional model that a Kalman filter could be applied to.

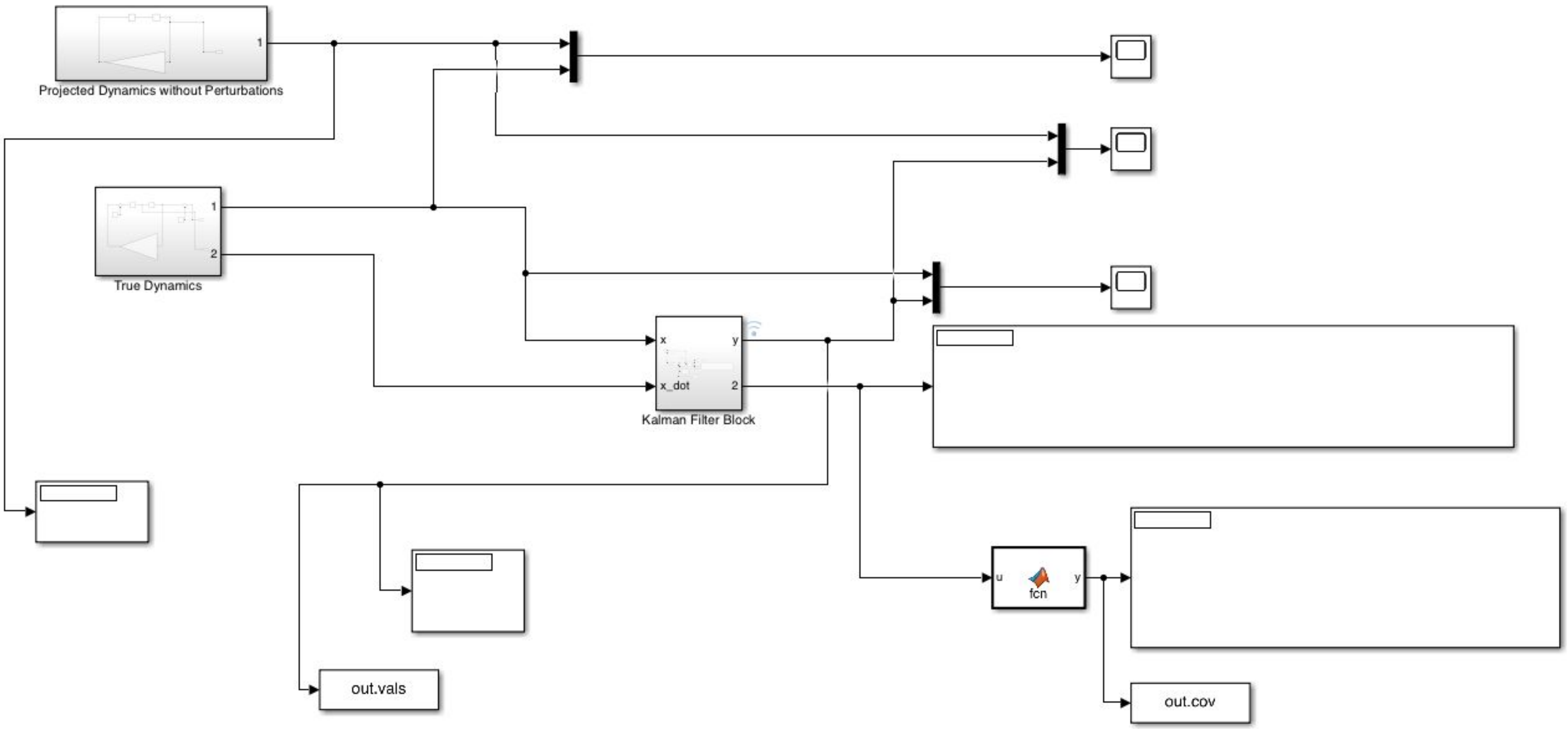
- The potential field was designed to impart an acceleration in the x direction, the y direction, and the z direction based on the location of the particle in 3D space.

- The accelerations were chosen to be:
  X_Acceleration= -2x+1y-2z
  Y_Acceleration= 4x-50y+2z
  Z_Acceleration= -2x-5y-4z
  This created a coupled system.

- The Kalman filter outputs the covariance matrix used to plot the error ellipsoid.

- The probability ellipsoid was visualized using the surfl MATLAB plotting tool.

Potential Field Block Diagram:



3D DRONE SIMULATION DEMO

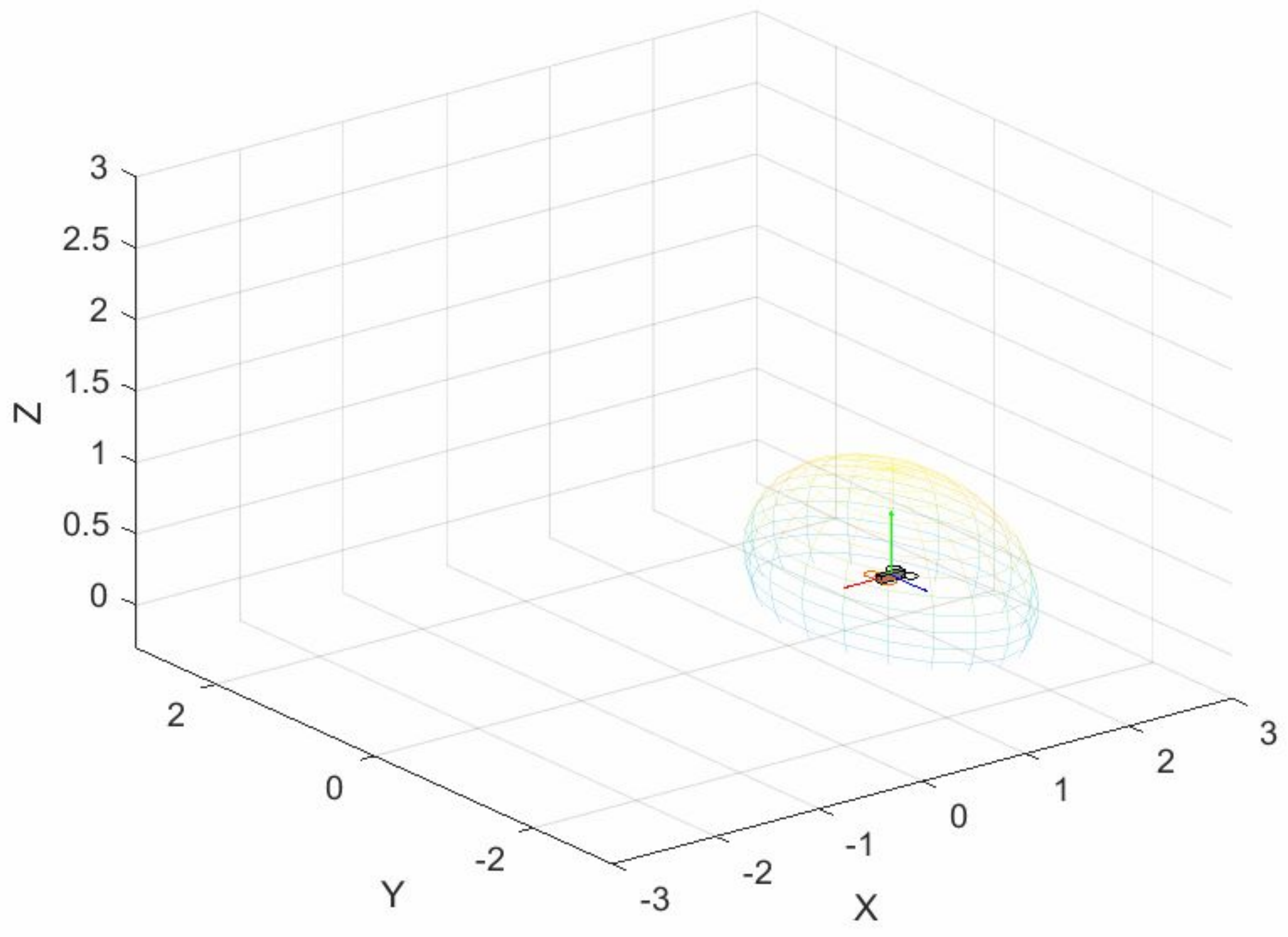
Simulation Model

- Model of drone dynamics based on newtonian physics for system prediction
- Simulated drone nominal trajectory is a circle
- PD controller implemented to follow the given trajectory
- Kalman filter combining model with simulated measurements from onboard instruments to determine position and covariance of position

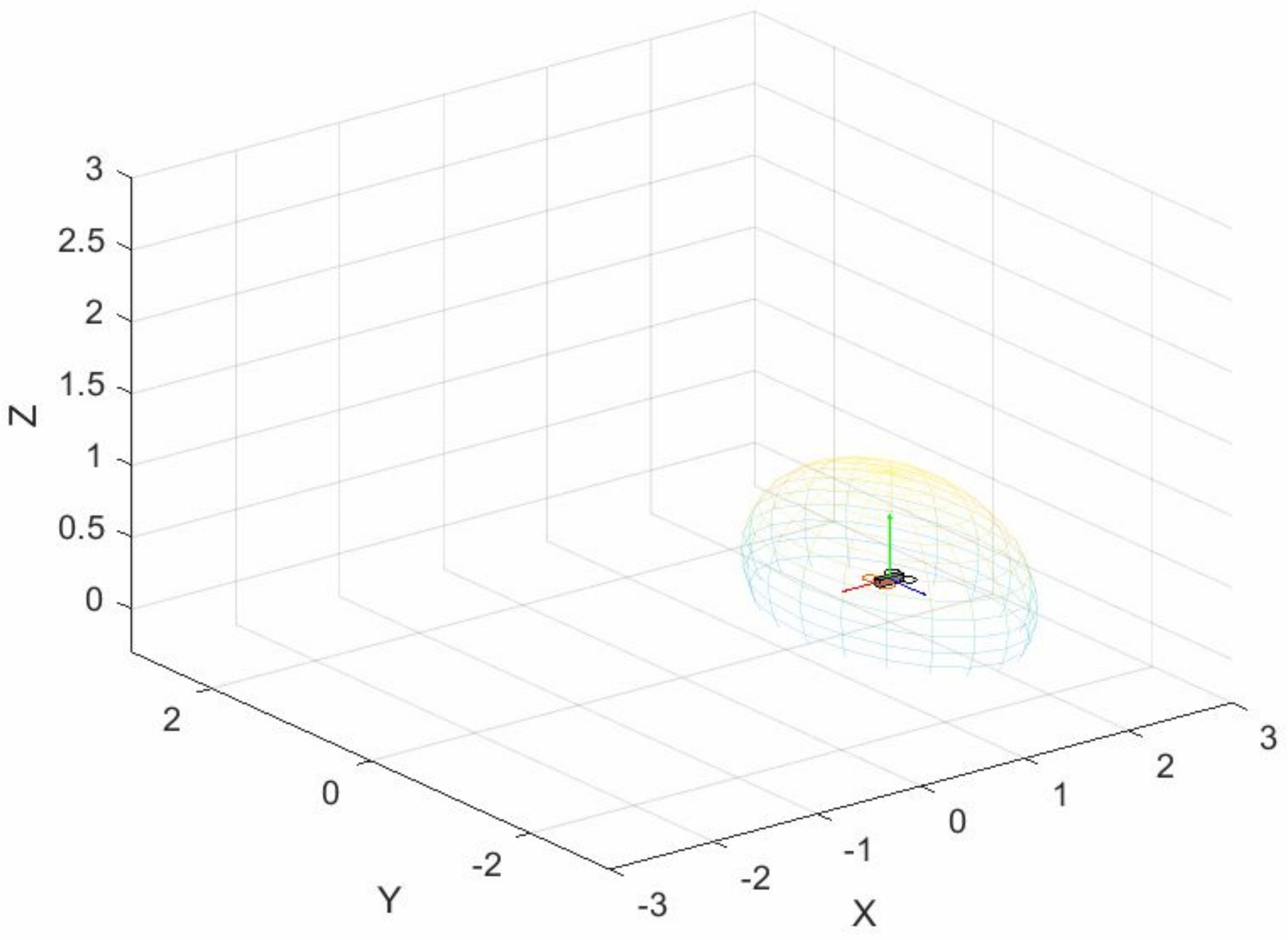
Simulated Sensors

- Camera using known points in environment; Modeled using a pinhole camera model
- IMU data; Accelerometer and gyroscope
- GPS positioning with two receiving antennas
- Error covariances of sensors based on spec sheets and variance modeled as gaussian

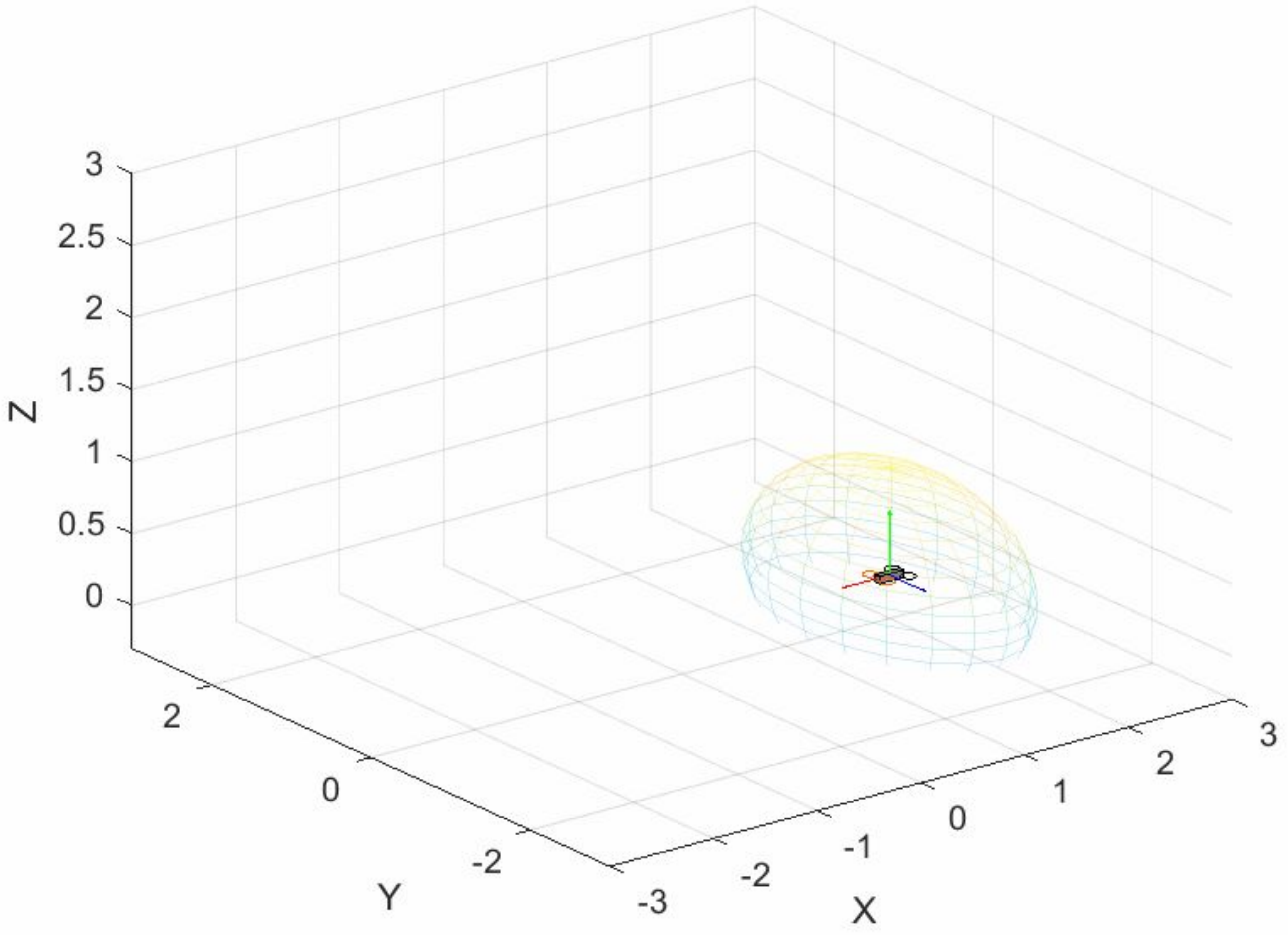
- Close to optimal proportional and derivative gains on controller
- Nice smooth flightpath



- Non-optimal gains
- Wonky flightpath



- Terrible gains
- Non-circular flightpath



Error Ellipsoid Generation

PROBABILITY ELLIPSOID

$$(\mathbf{x}-\overline{\mathbf{x}})^T P^{-1}(\mathbf{x}-\overline{\mathbf{x}}) = \ell^2 \quad (1)$$

$$U = [\mathbf{u}_1,\ \mathbf{u}_2, \ldots,\ \mathbf{u}_n]_{n\times n} \quad (2)$$

$$U^T P U = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} = D[\lambda_1,\ \lambda_2, \ldots,\ \lambda_n] \quad (3)$$

$$\mathbf{x}' = U^T \mathbf{x} \quad (4)$$

$$\begin{aligned} P' &\equiv E[(\mathbf{x}'-\overline{\mathbf{x}}')(\mathbf{x}'-\overline{\mathbf{x}}')^T] \\ &= U^T E[(\mathbf{x}-\overline{\mathbf{x}})(\mathbf{x}-\overline{\mathbf{x}})^T]\, U \\ &= U^T P U = D[\lambda_1 \ldots \lambda_n]. \end{aligned} \quad (5)$$

$$\Delta\mathbf{x} \equiv \hat{\mathbf{x}} - \mathbf{x} \equiv [\tilde{x}\ \tilde{y}\ \tilde{z}]^T \quad (6)$$

$$P = E[\Delta\mathbf{x}\Delta\mathbf{x}^T] \quad (7)$$

$$[\tilde{x}\ \tilde{y}\ \tilde{z}]P^{-1}\begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{bmatrix} = \ell^2 \quad (8)$$

$$\begin{bmatrix} \tilde{x}' \\ \tilde{y}' \\ \tilde{z}' \end{bmatrix} = U^T \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{bmatrix} \quad (9)$$

$$P' = U^T P U \quad (10)$$

$$[\tilde{x}'\ \tilde{y}'\ \tilde{z}']\begin{bmatrix} 1/\lambda_1 & & \\ & 1/\lambda_2 & \\ & & 1/\lambda_3 \end{bmatrix}\begin{bmatrix} \tilde{x}' \\ \tilde{y}' \\ \tilde{z}' \end{bmatrix} = \ell^2 \quad (11)$$

$$\frac{\tilde{x}'^2}{\lambda_1} + \frac{\tilde{y}'^2}{\lambda_2} + \frac{\tilde{z}'^2}{\lambda_3} = \ell^2 \quad (12)$$

Probability Ellipsoid Equation

$\ell = 3 \rightarrow 3\sigma$ probability ellipsoid with 97.1% confidence

PROBABILITY ELLIPSOID

$\ell = 3 \rightarrow 3\sigma$ Probability ellipsoid with 97.1% confidence.

Assumption: Trivariate normal distribution

$$P(x_1, x_2, x_3) = \frac{e^{-w/[2(\rho_{12}^2+\rho_{13}^2+\rho_{23}^2-2\rho_{12}\rho_{13}\rho_{23}-1)]}}{2\sqrt{2}\,\pi^{3/2}\sqrt{1-(\rho_{12}^2+\rho_{13}^2+\rho_{23}^2)+2\rho_{12}\rho_{13}\rho_{23}}} \quad (13)$$

$$w = x_1^2(\rho_{23}^2-1)+x_2^2(\rho_{13}^2-1)+x_3^2(\rho_{12}^2-1)+2[x_1x_2(\rho_{12}-\rho_{13}\rho_{23})+x_1x_3(\rho_{13}-\rho_{12}\rho_{23})+x_2x_3(\rho_{23}-\rho_{12}\rho_{13})] \quad (14)$$

PROBABILITY ELLIPSOID

$$\frac{\tilde{x}'^2}{\lambda_1} + \frac{\tilde{y}'^2}{\lambda_2} + \frac{\tilde{z}'^2}{\lambda_3} = \ell^2 \quad (15)$$

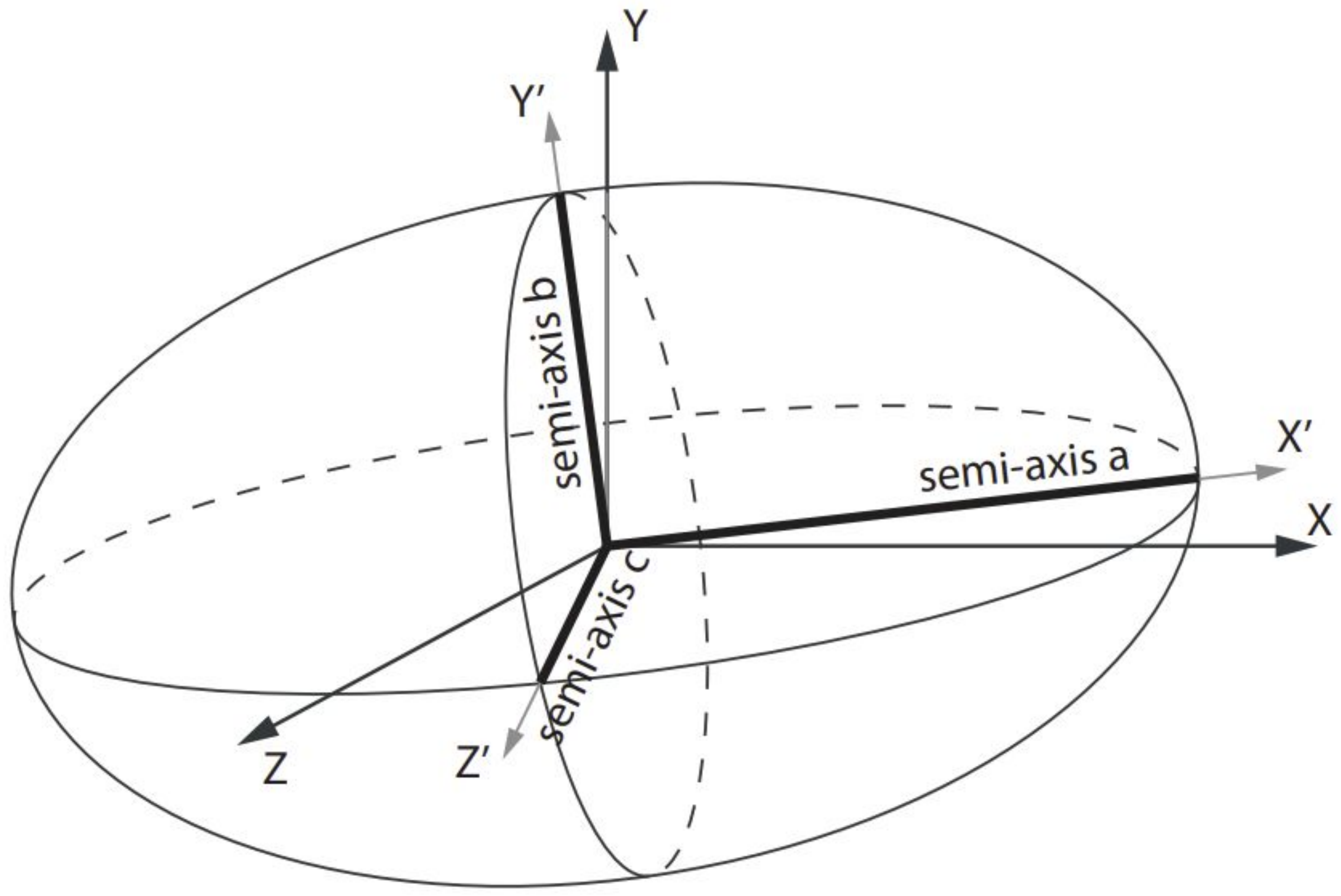
Solve for one variable,

set other two equal to 0

$$a = \tilde{x}' = \sqrt{\lambda_1 * \ell^2}$$

$$b = \tilde{y}' = \sqrt{\lambda_2 * \ell^2} \quad (16)$$

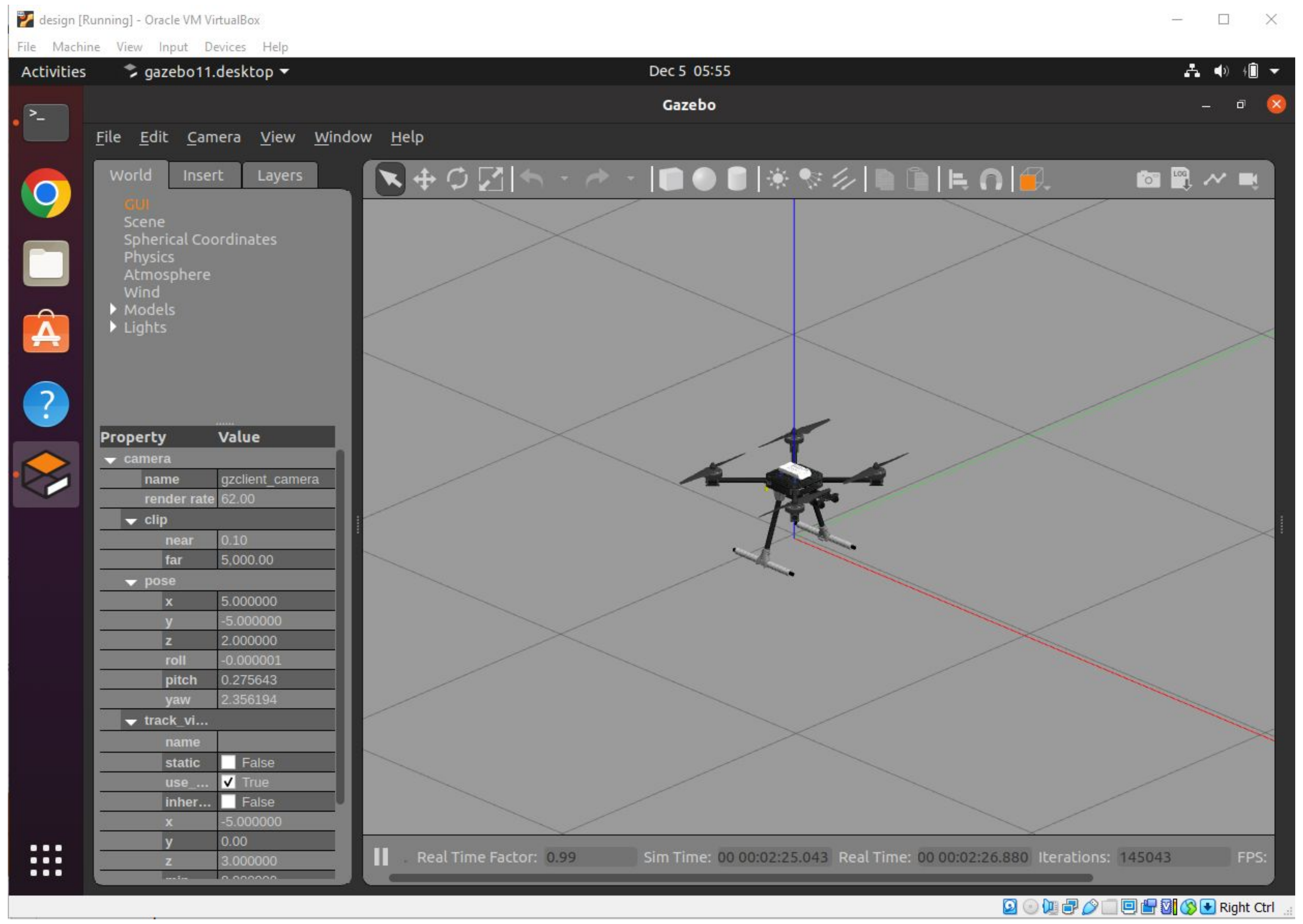
$$c = \tilde{z}' = \sqrt{\lambda_3 * \ell^2}$$



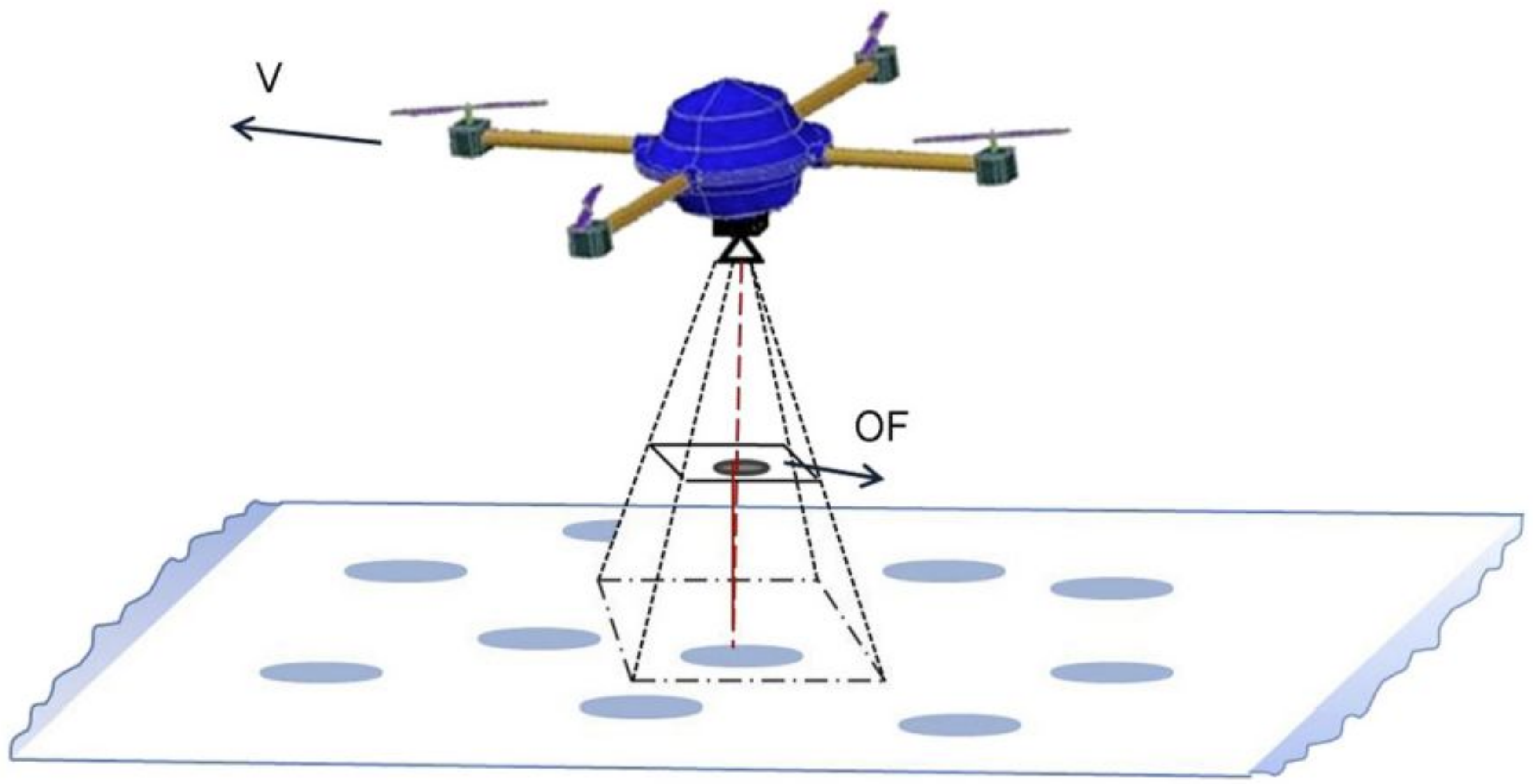
NEXT STEPS

GAZEBO

- Develop a 3D model simulation with a time-varying error ellipse to account for the more realistic conditions of variable noise
- Research how to apply the extended kalman filter to a 3D system
- Develop a functional simulation that can estimate the drone state and visualize the probability ellipsoid based on flight and sensor inputs



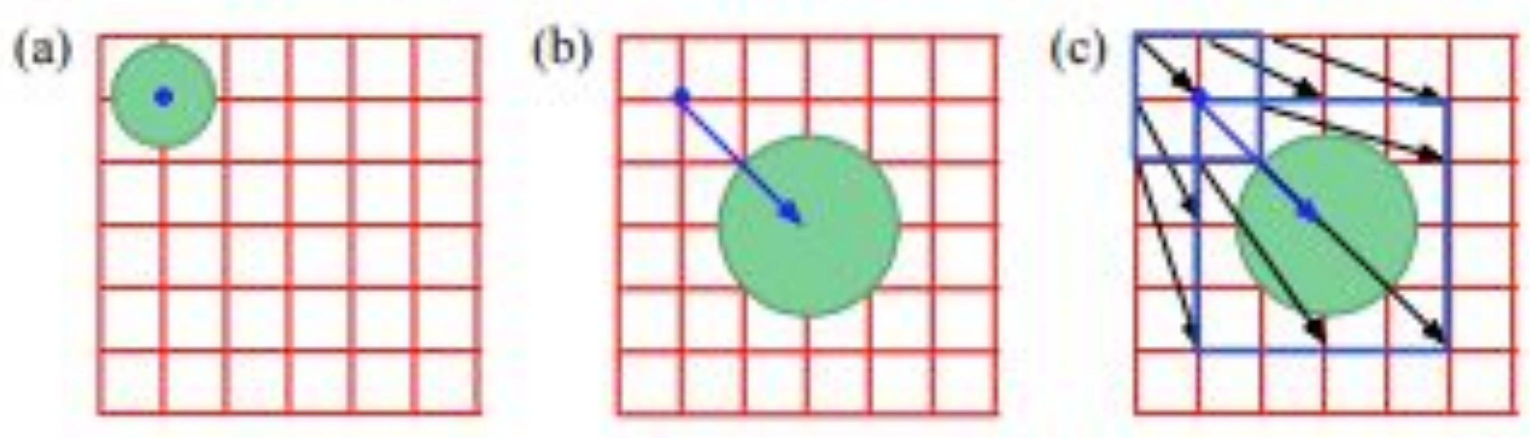
OPTICAL FLOW



- Method of calculating relative 2D translational motion
- Utilizes time-ordered images from camera to calculate motion between frames
    - Image converted to grayscale → Obtain gradient of points → assume intensity of each point is unchanging

$$I_x v_{x_{im}} + I_y v_{y_{im}} + I_t = 0$$

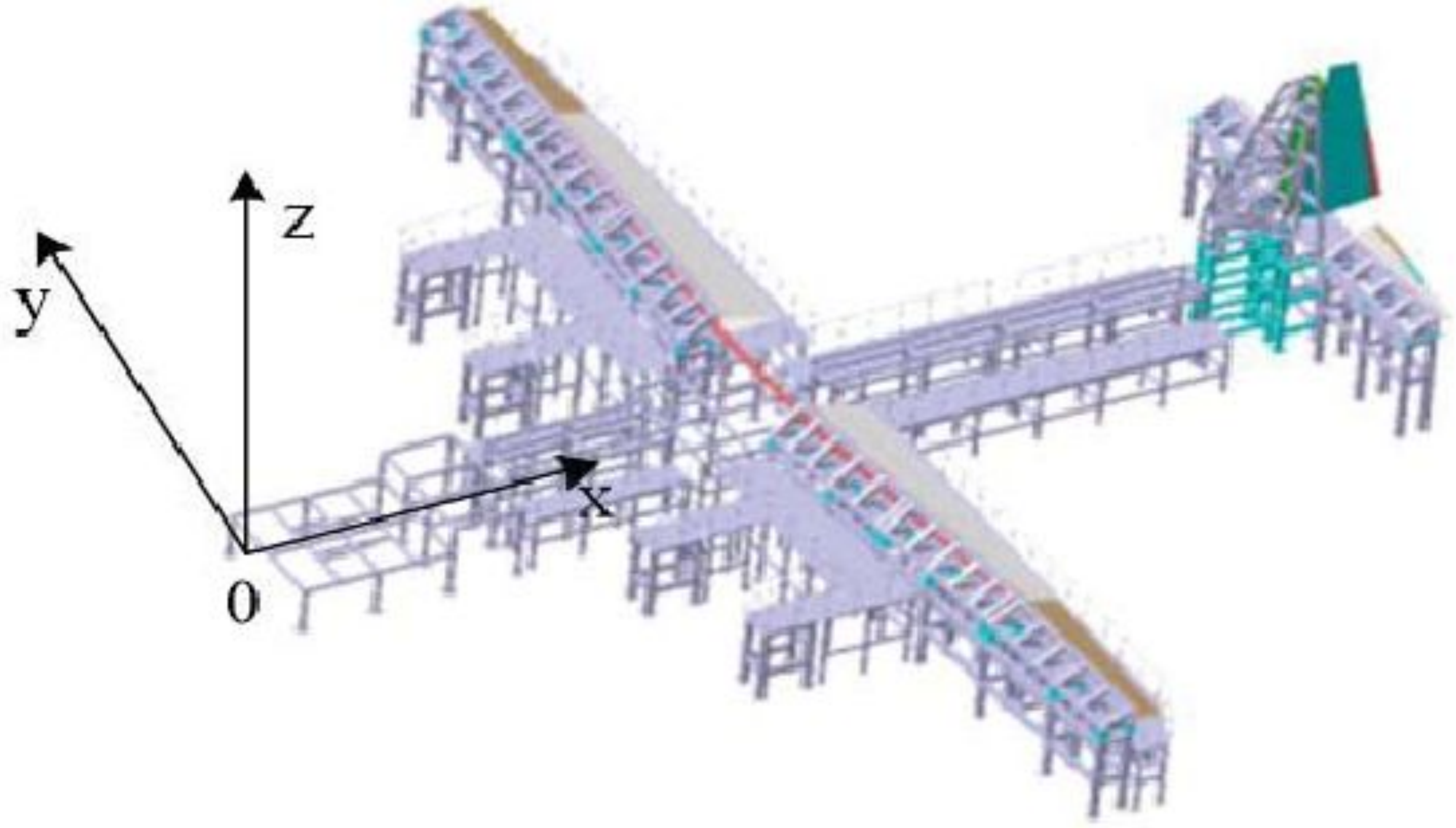
OPTICAL EXPANSION*



- Optical flow only works well when non-complex 2D motion occurs
- Optical expansion is a possible technique for obtaining 3D scene flow, which may provide meaningful directions of 3D movement
    - Similar process to optical flow, but analyzes the change in the perceived size of objects to obtain motion in depth
    - Still suffers from scale ambiguity

* If necessary or if time allows

WOOD BIRD SETUP



- Allows for systems testing by multiple groups
- Integrates all sensors and other electronics to be used on the drone

GANTT CHART

TASK TITLE	START ESTIMATION	TENTATIVE DUE DATE	COMPLETE (?)	1/9	1/16	1/23	1/30	2/6	2/13	2/20	2/27	3/6	3/20	3/27	4/3	4/10	4/17	4/24
Simulation																		
Dynamic Model	1/9/2023	2/6/2023																
Upload specs to model parameters	1/30/2023	2/6/2023																
Kalman Filter for Dynamic Model	1/30/2023	2/20/2023																
Sensor Models	2/6/2023	2/20/2023																
Simulink Flight	2/20/2023	3/20/2023																
MATLAB Simulation	3/13/2023	4/24/2023																
Estimation																		
Research Error Ellipsoid Methods	1/9/2023	1/30/2023																
Implement Relevant Error Ellipsoid to Model	2/13/2023	3/6/2023																
Implement Error Ellipsoid in MATLAB	3/27/2023	4/24/2023																
Flight Software																		
Manual Flight PX4 Installation	1/9/2023	2/6/2023																
QGroundControl Setup	1/30/2023	2/20/2023																
Test Computer Vision Software	2/27/2023	3/20/2023																
Research Obstacle Avoidance Algorithms	3/20/2023	4/24/2023																

FUTURE WORK

- Develop a 3D model simulation with a time-varying error ellipse to account for the more realistic conditions of variable noise
- Develop a functional Gazebo simulation that can estimate the drone state and visualize the probability ellipsoid based on flight and sensor inputs
- Implement code onto physical drone
  - Flight Software (PX4)
  - Setup Ground Station Laptop
- Investigate alternate flight control

THANK YOU.

USRC - Simulation / Estimation

- Izaac Facundo
- Nicholas Franken
- Jeremy Morrison
- Neel Pandey
- Preston Thomas
- William Wang

The University of Texas at Austin
Aerospace Engineering and Engineering Mechanics
Cockrell School of Engineering

References:

[1] “Plot 3D probability ellipsoid,” https://kittipatkampa.wordpress.com. https://kittipatkampa.wordpress.com/2011/08/04/plot-3d-ellipsoid/