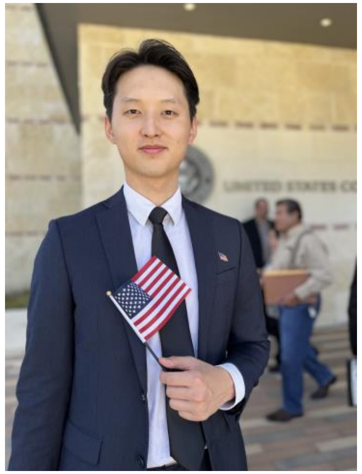
Bonsuck Koo

New Graduate Rotation Program– Guidance, Navigation & Control

TEXAS
The University of Texas at Austin

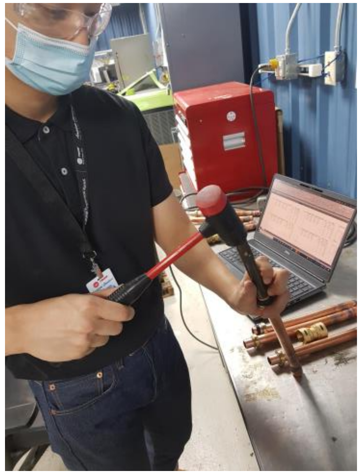
Introduction

- Name: Bonsuck Koo
- School: University of Texas at Austin
- Degree: Integrated Master and Bachelor of Science in Mechanical Engineering (2024 Dec.)
- Hobbies: Visiting National Parks
- Obtained American Citizenship in July 2023



Experience overview

- NGC engineering intern
  - Sandia National Laboratories (Current)
- GNC engineering Intern
  - Blue Origin (2023 Fall)
- Mechanical Engineering Intern
  - Samsung Austin Semiconductors (2022 Summer)
- System Engineering Intern
  - Trane Technologies (2021 Summer and Fall)
- Mandatory Korean Military service (2018-2020)



Passion for our Mission

Why Blue Origin?

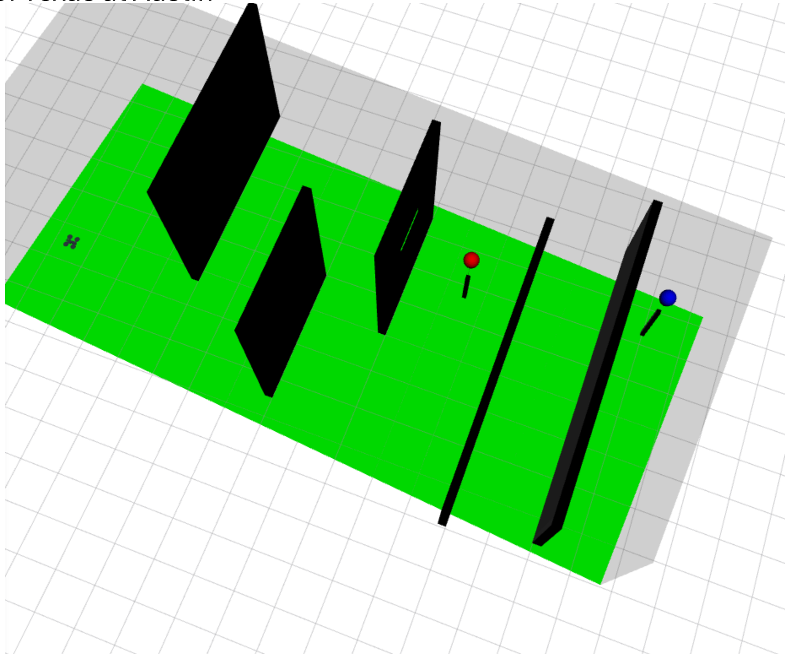
Project

Aerial Robotics (Drone)

Aerial Robotics (Drone)

- 2024 Spring (Last Semester)
- Team Competition
- Goal:
  - Create an algorithm to find the fastest routes to predetermined targets
  - The final algorithm is implemented in a high-fidelity simulation tool

Source: ASE 497W Todd E. Humphreys; University of Texas at Austin



My contributions

- Develop 6-DOF simulation in MATLAB
- Develop A* algorithm in C++

Aerial Robotics (Drone)

Quadrotor Dynamics

$$m\dot{\boldsymbol{r}}_{\mathrm{I}} = m\boldsymbol{v}_{\mathrm{I}}$$

Gravity

Force from the Fan

Disturbance

Wind Resistance

$$m\dot{\boldsymbol{v}}_{\mathrm{I}} = m\ddot{\boldsymbol{r}}_{\mathrm{I}} = -mg\boldsymbol{z}_{\mathrm{I}} + \sum_{i=1}^{4}\boldsymbol{F}_{i\mathrm{I}} + \boldsymbol{d}_{\mathrm{I}} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + R_{\mathrm{BI}}^{\mathrm{T}}\sum_{i=1}^{4}\begin{bmatrix} 0 \\ 0 \\ F_i \end{bmatrix} + \boldsymbol{d}_{\mathrm{I}} - d_a \boldsymbol{v}_{\boldsymbol{I}}^{\boldsymbol{u}}$$

$$\dot{\boldsymbol{\omega}}_{\mathrm{B}} = J^{-1}\left(\boldsymbol{N}_{\mathrm{B}} - [\boldsymbol{\omega}_{\mathrm{B}}\times]J\boldsymbol{\omega}_{\mathrm{B}}\right) \quad \text{where} \quad \boldsymbol{N}_{\mathrm{B}} = \sum_{i=1}^{4}\left(\boldsymbol{N}_{i\mathrm{B}} + \boldsymbol{r}_{i\mathrm{B}} \times \boldsymbol{F}_{i\mathrm{B}}\right)$$

Torque from propeller acting against the air

Torque from upward thrust

$$\dot{R}_{\mathrm{BI}} = -[\boldsymbol{\omega}_{\mathrm{B}}\times]R_{\mathrm{BI}}$$

Source: ASE 497W Todd E. Humphreys; University of Texas at Austin

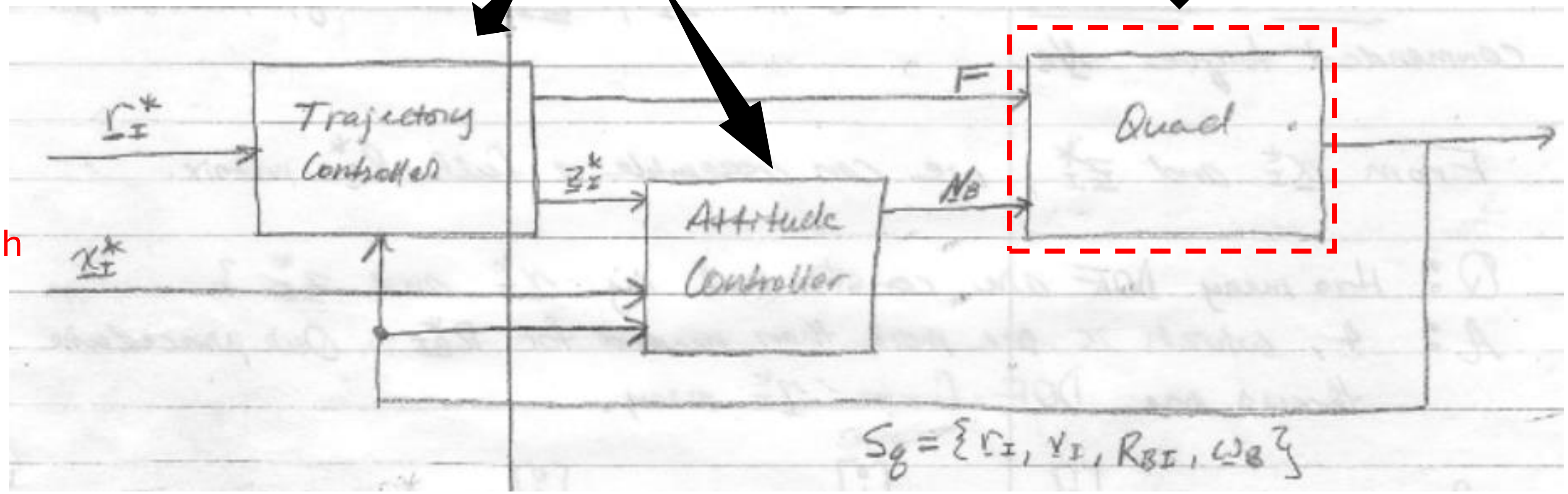
Aerial Robotics (Drone)

PD controller

PD Controller

Plant

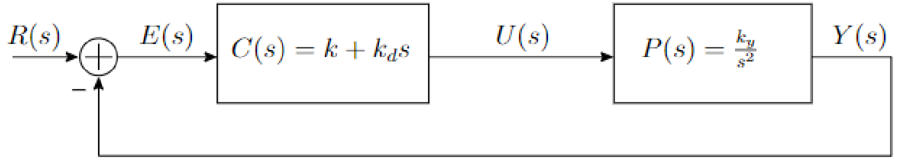
Reference or desired states are denoted with a star

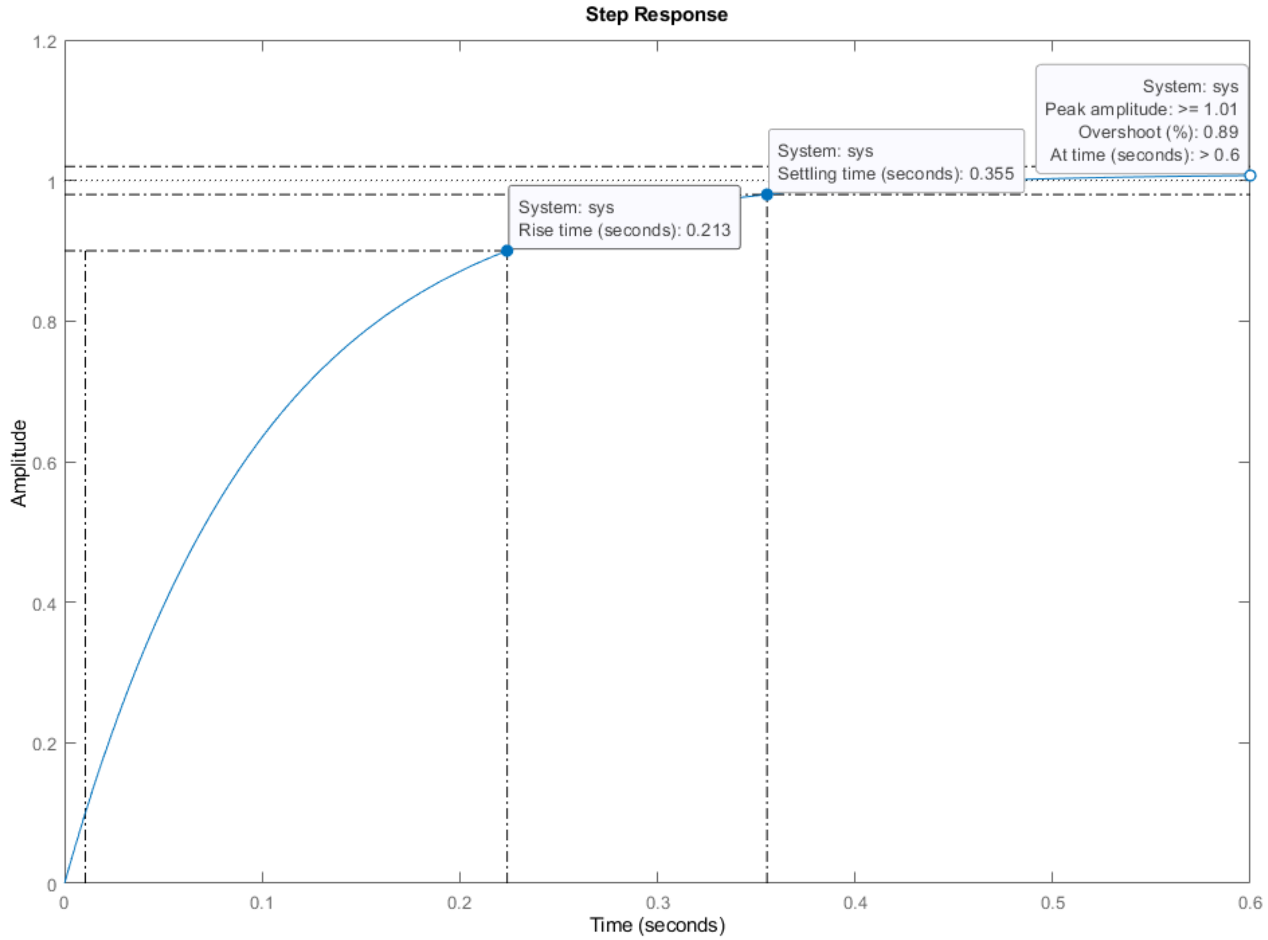


Source: ASE 497W Todd E. Humphreys; University of Texas at Austin

Aerial Robotics (Drone)

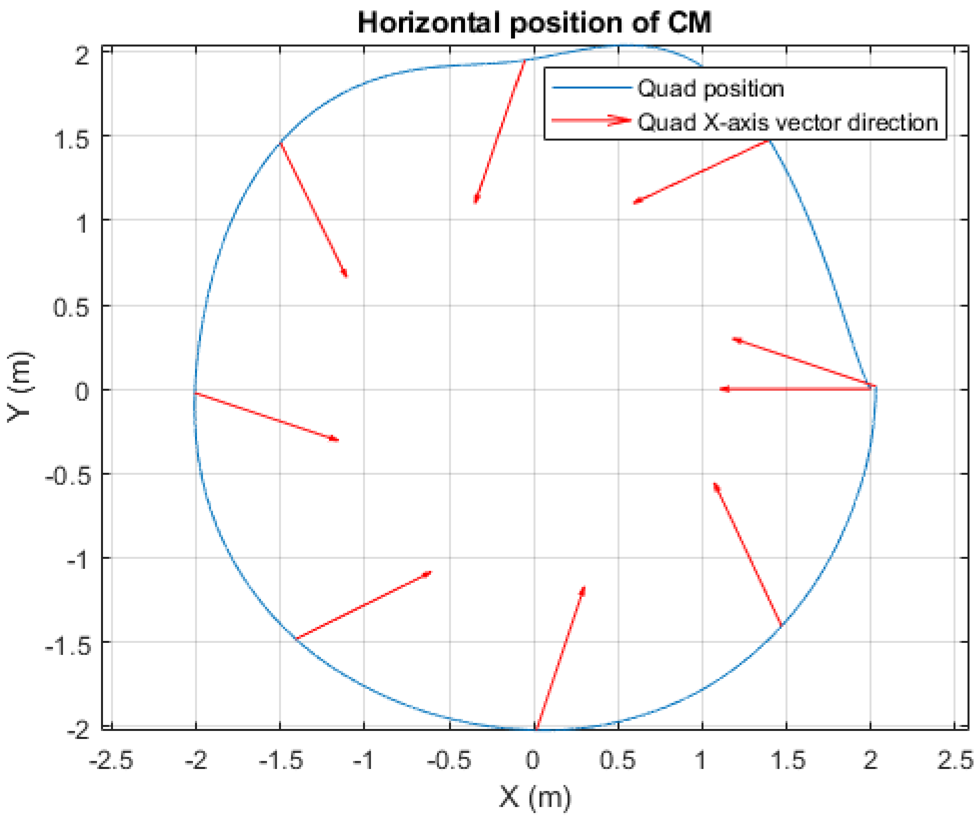
PD controller

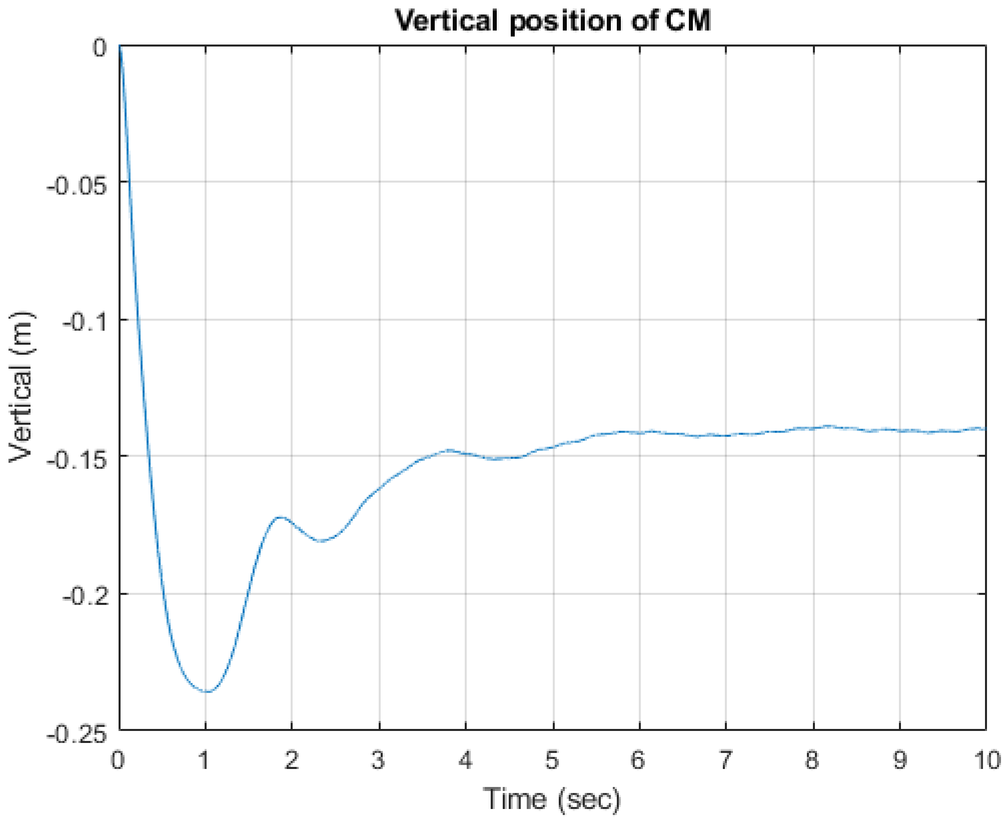




Aerial Robotics (Drone)

PD controller





Attitude Controller PD Gains

```
%% PD gains
K  = diag([0.05 0.25 0.05]);
Kd = diag([0.5 0.25 0.05]);
```

Trajectory Controller PD Gains

```
%% PD gains
kr = 1.5;
kd = 5;
```

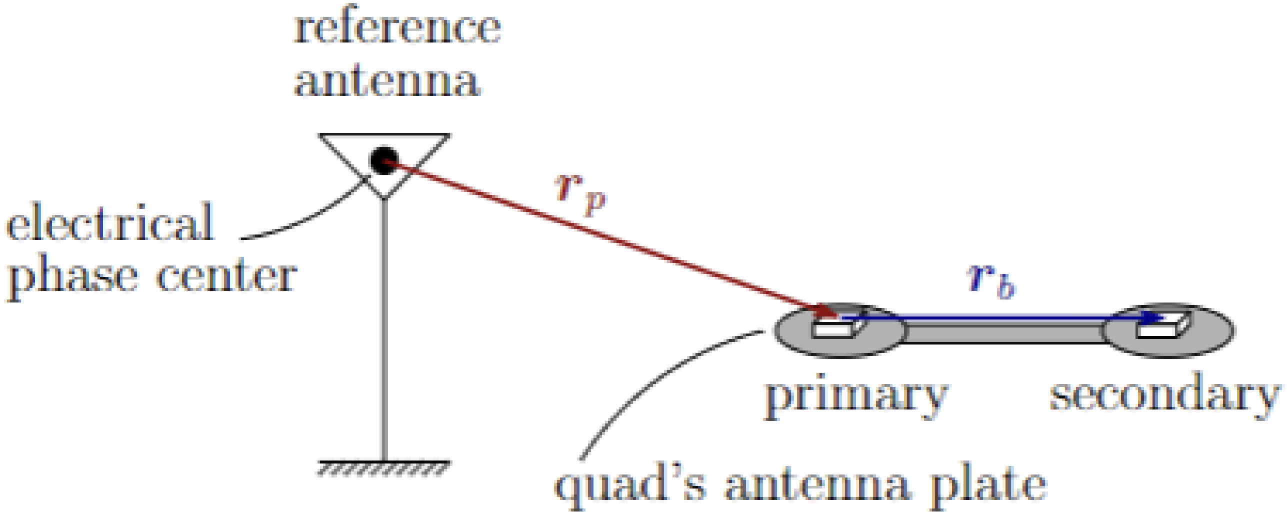
Aerial Robotics (Drone)

Sensor Modeling

- GNSS Measurements

$$\tilde{r}_{pG}(t_k) = r_{pG}(t_k) + w_{pG}(t_k)$$

```
rpI = S.statek.rI + RIB*P.sensorParams.raB(:,1);
rpG = RIG'*rpI;
rpGtilde = rpG + RPa'*randn(3,1);
```



Source: ASE 497W Todd E. Humphreys; University of Texas at Austin

Source: ASE 497W Todd E. Humphreys; University of Texas at Austin

Aerial Robotics (Drone)

Sensor Modeling

- IMU
  - Accelerometer Measurement Model

Accelerometer Bias

Acceleration

$$\tilde{f}_{\mathrm{B}} = R_{\mathrm{BI}}(a_{\mathrm{I}} + g e_3) + b_a + v_a$$

Gravity Accelerometer Noise

$$R_{\mathrm{BI}} a_{\mathrm{I}} = \underbrace{R_{\mathrm{BI}} \ddot{R}_{\mathrm{I}}}_{\text{accel. of B origin}} + \cancel{\underbrace{\ddot{l}_{\mathrm{B}}}_{\text{rel. accel.}}} + \cancel{\underbrace{(\dot{\omega}_{\mathrm{B}} \times l_{\mathrm{B}})}_{\text{Euler}}} + \cancel{\underbrace{2(\omega_{\mathrm{B}} \times \dot{l}_{\mathrm{B}})}_{\text{Coriolis}}} + \cancel{\underbrace{\omega_{\mathrm{B}} \times (\omega_{\mathrm{B}} \times l_{\mathrm{B}})}_{\text{centripetal}}}$$

  - Rate Gyro Measurement Model

Gyro Bias

$$\tilde{\omega}_{\mathrm{B}} = \omega_{\mathrm{B}} + b_g + v_g$$ Gyro Noise

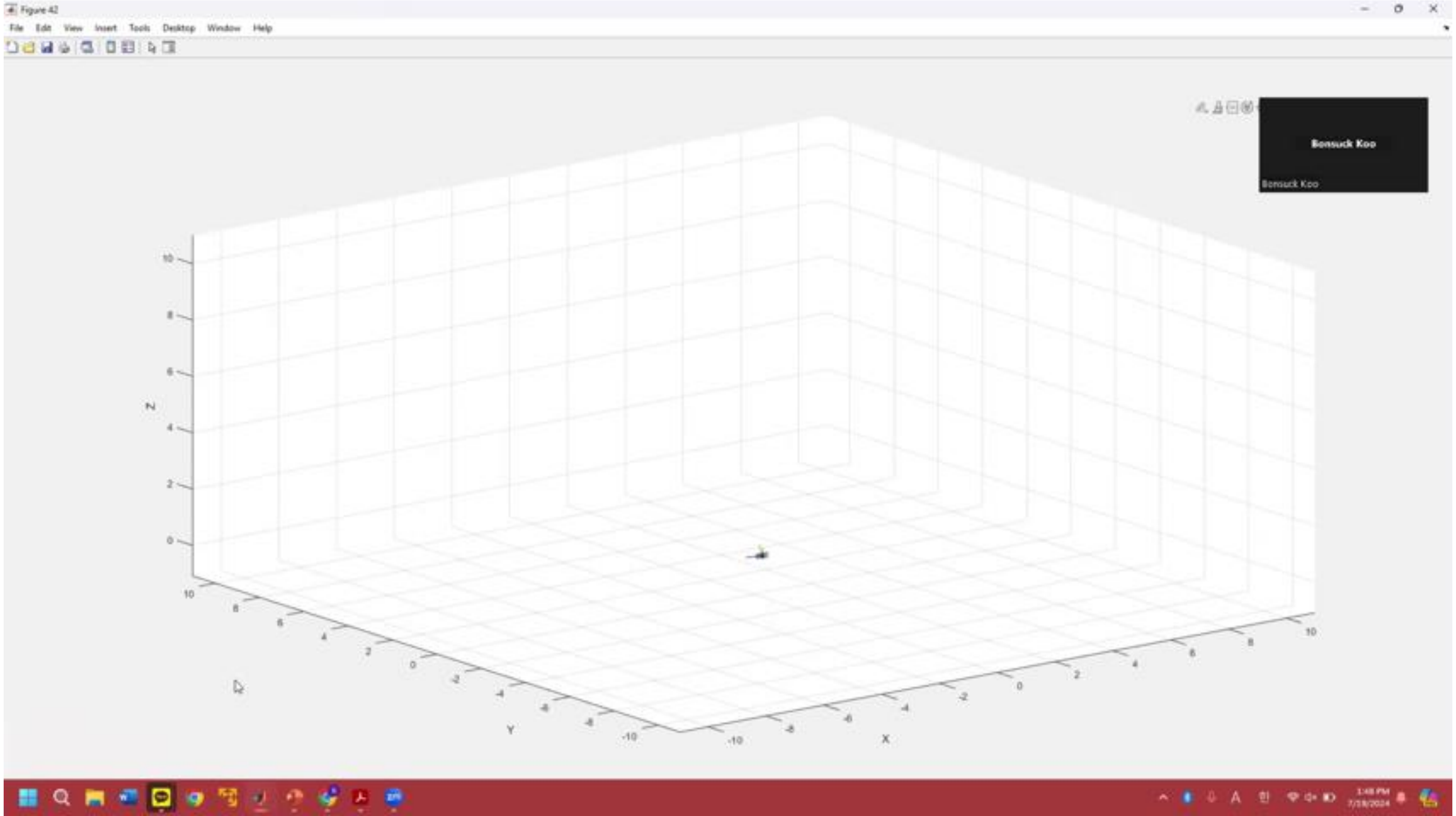
Angular Velocity

Aerial Robotics (Drone)

- Additional Modeling
  - Camera
  - Unscented Kalman Filter
- Full MATLAB simulation
  - Goal:
    - Follow reference trajectory
      - Determined by A* algorithm in C++
    - Maintain 0 m altitude

Aerial Robotics (Drone)

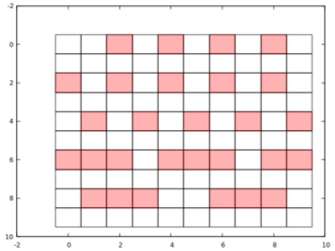
Completed MATLAB Simulation



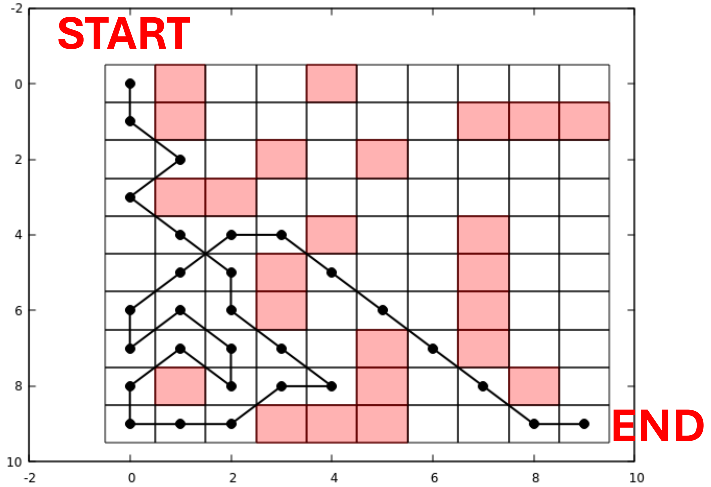
Aerial Robotics (Drone)

C++: Path Finding Algorithm

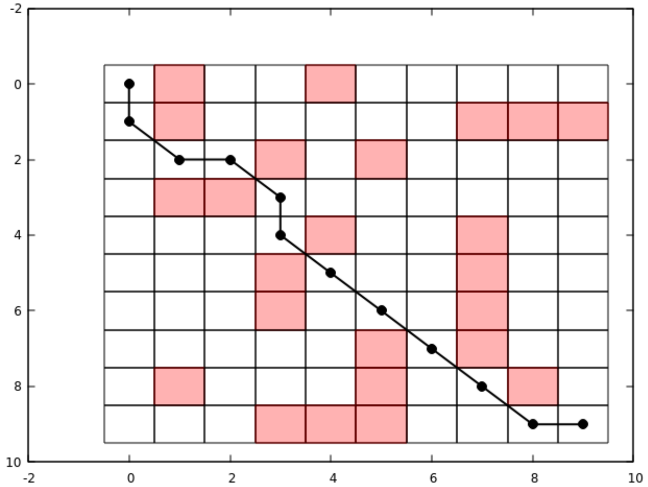
- Comparison of three different path finding algorithms in C++
  - Selection Criteria:
    1. Number of Nodes Explored
    2. Number of Nodes in path



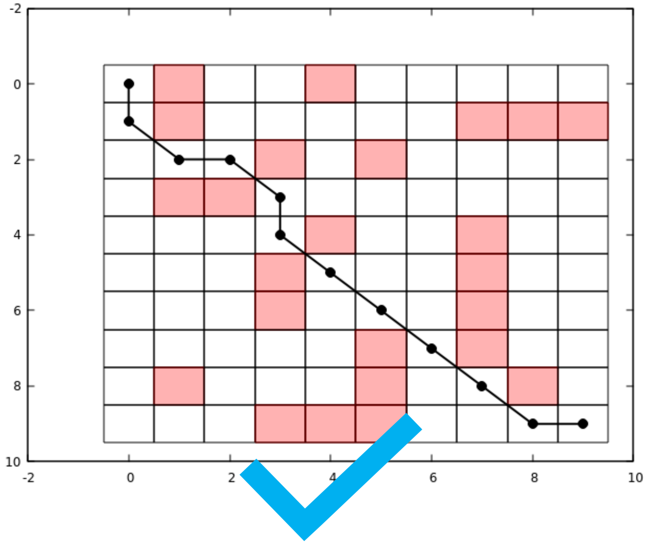
A field gridded into cells and mapped into nodes. Nodes are at the center of each cell. Red cells represent obstacles.



Depth First Search (DFS)



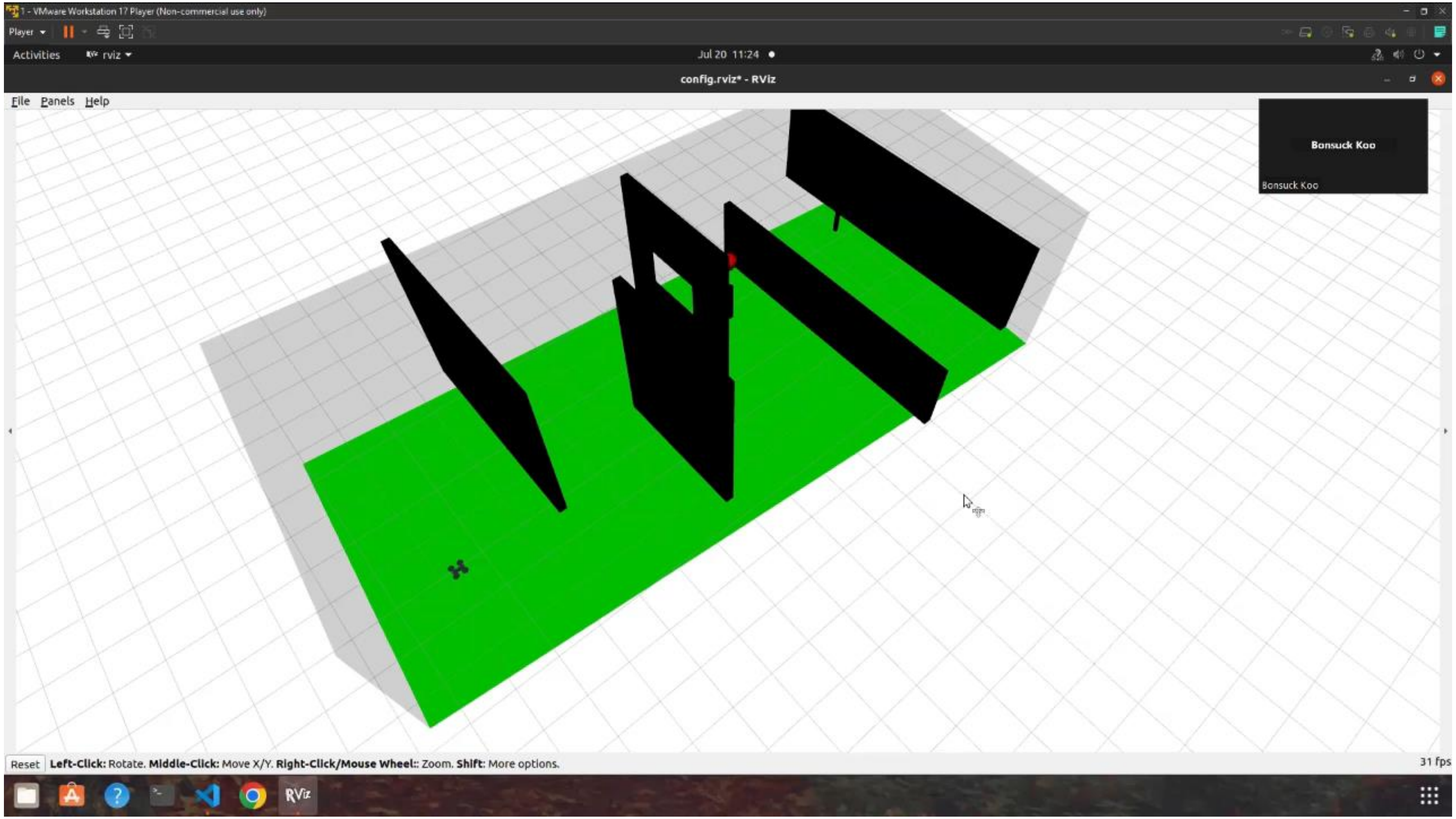
Dijkstra's



A*

Aerial Robotics (Drone)

C++: Path Finding Algorithm



Aerial Robotics (Drone) Team Collaboration

Communication and Feedback

GIT

Experience at Blue helped me lead the Team to utilize GitLab

Summary of Drone Project

Situation

A competition to pop the balloons with a drone in the least amount of time

Task

Develop MATLAB simulation

Develop path finding software for the drone

Action

Apply knowledge in dynamics, classical control theory, navigation sensors (accelerometer, gyroscopes, GNSS receiver) for MATLAB simulation

Implemented A* method in C++ for path finding algorithm

Feedback through active Communication

Git to collaborate

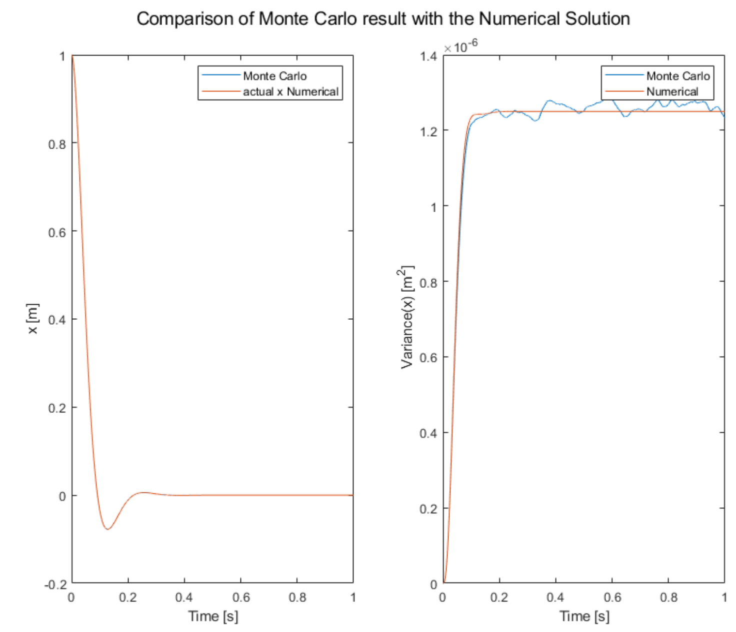
Result

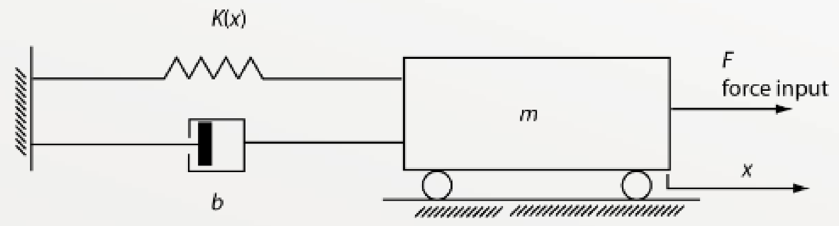
2^{nd} place!

Q&A

Experience overview

- Relevant Courses:
    - Automated Control Systems
    - Dynamic Systems and Control
    - Spacecraft Dynamics
    - Aircraft Dynamics
    - Stochastic Estimation and control
    - Aerial Robotics
    - GPS Signal Processing





a. Show that the stochastic differential equations of motion can be expressed as:

$$dx = \frac{p}{m}dt$$

$$dp = \left(-Kx - \left(\frac{b}{m}\right)p\right)dt + d\beta$$