OPTIMAL CONTROL FOR DRONES

LiangYu Chen BUID : U32606850

Problem Statement :

In this project, I chose optimal control for drone because its importance has gradually grown year by year. How could we make sure the drones would avoid the obstacles accurately and complete the mission with minimum cost. The answer is to control the drones in the optimal way. There are lots of things that involved with control. For instance, driving a car, operate multifunction robots, etc. We first need to define what is our goal and list some constraints. Constraints could be avoiding the obstacles, saving energy according to the current condition, and others based on different scenarios. Control plays an important role here. For example, what kind of the control input should we tell to the sweeping robot in order to clean the floor without bumping into the furnitures. Finding the optimal control is significant since there may be many control inputs that can achieve the goal. However, we want to pick the optimal one because it can minimize out cost or maximize our profit, depends on how we define the problem.

Application :

Many fields contain the application of control. I'll introduce some examples from the course, SE701, I took last semester. A simple one is the fish harvest model. Let's say our goal is to maximize the return over a fixed interval [0,T], our return function $r(t) = \gamma e(t)x(t) - ce(t)$. Where x(t) is the fish population and e(t) presents our effort to raise the fish. We also have a constraint of population change rate $\dot{x} = \alpha e(t) - ce(t), e(t) \in [0,1] \forall t$. Here comes a simple control problem, how much should we put the effort to get max return? The effort e(t) is the input and the return r(t) is cost function and $\dot{x}$(t) is the population dynamics. In real world, of course we get lots more constraints to find the optimal solution. Another example I want to explain how useful control can be. Let's say Gout is a disease caused by excess uric acid in the blood. We denoted the excess as x(t) varies with time and surely want the x(t) to decrease to zero as fast as possible. The method to decrease the excess is to take medicine (the dosage of the medicine is the input u(t)). However, the medicine is expensive and would have the side effects. We are given the model of excess rate $\dot{x}(t) = 1 - x(t) - u(t)$ and our cost function is the dosage of medicine $\min_u J(u) = \int_0^{t_1} \frac{1}{2}(k^2 + u^2(t))dt$. Here k is a constant to weight the importance of the amount of drug against to the terminal time. We may also be given some boundary conditions and more constraints to solve the optimal control problem. For this project. I'll focus on the optimal control of drones since drones will be applied in our daily life in the future in different aspects.

Literature Review

I picked some papers to read before I jump into the optimal control problem of the drones. The following would be some abstract of these papers. The first paper is **Optimal**

Control Strategies for Load Carrying Drones[1]. This paper control the drones with linear quadratic control (LQR) and model predictive control (MPC) respectively and compared advantages and disadvantages of both methods. The objective is to maintain the ball in the equilibrium point and regulate the drones' altitudes. The control can be divided into two stages. The high-level control decides the force to lift an altitude of the drones and ball and the low-level control is responsible for stabilizing the drones. Figure(1) is the control scheme for the problem.

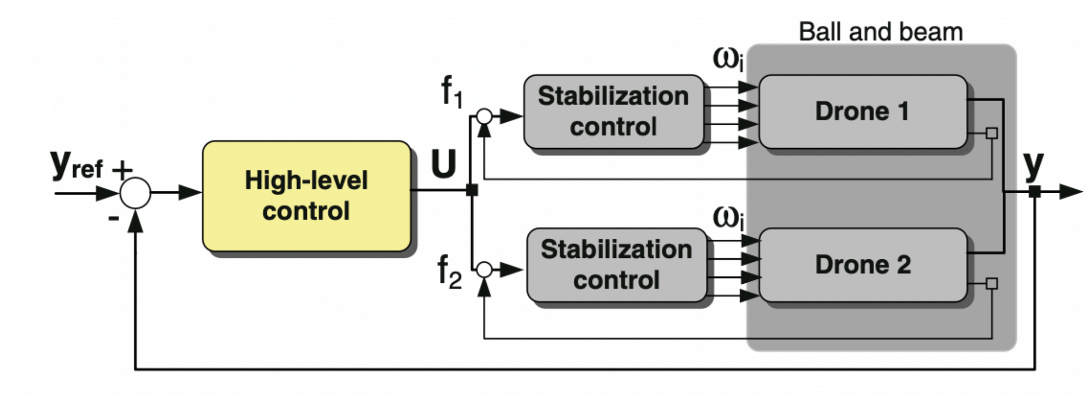
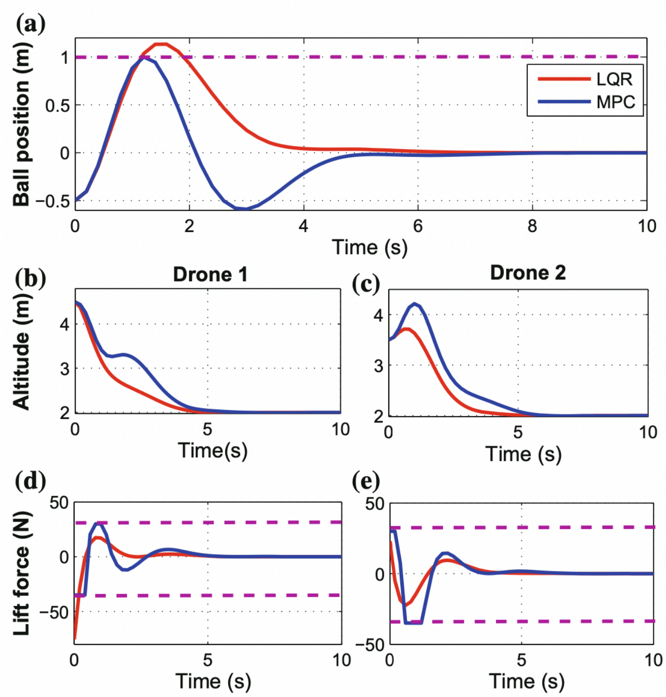


Figure (1) : Control scheme of the problem. Credit to [1]

LQR can solve the optimal problem with fast solving time and low computational cost, while MPC process the constraints and prediction well. The simulation result is in figure(2).



Figure(2) : Simulation result of different methods. Credit to [1]

Through observation, we can conclude that LQR responded faster than MPC for ball position. However, LQR exceeded the constraint that the ball position should not be higher than 1m. MPC, although responded to zero position a little slower, it did well while dealing with the constraints. For altitude of the two drones in figure 2 (b)(c), both LQR and MPC controlled well, and MPC still responded a bit slower. As for lift force, we can found similar result that LQR had shorter execution time, but violated the constraint that the lift force shouldn't exceed 50N. On the other side, MPC dealt with the constraint properly.

In the paper, they also simulated the situation with disturbances to compare two controllers. The result is shown in figure(3).

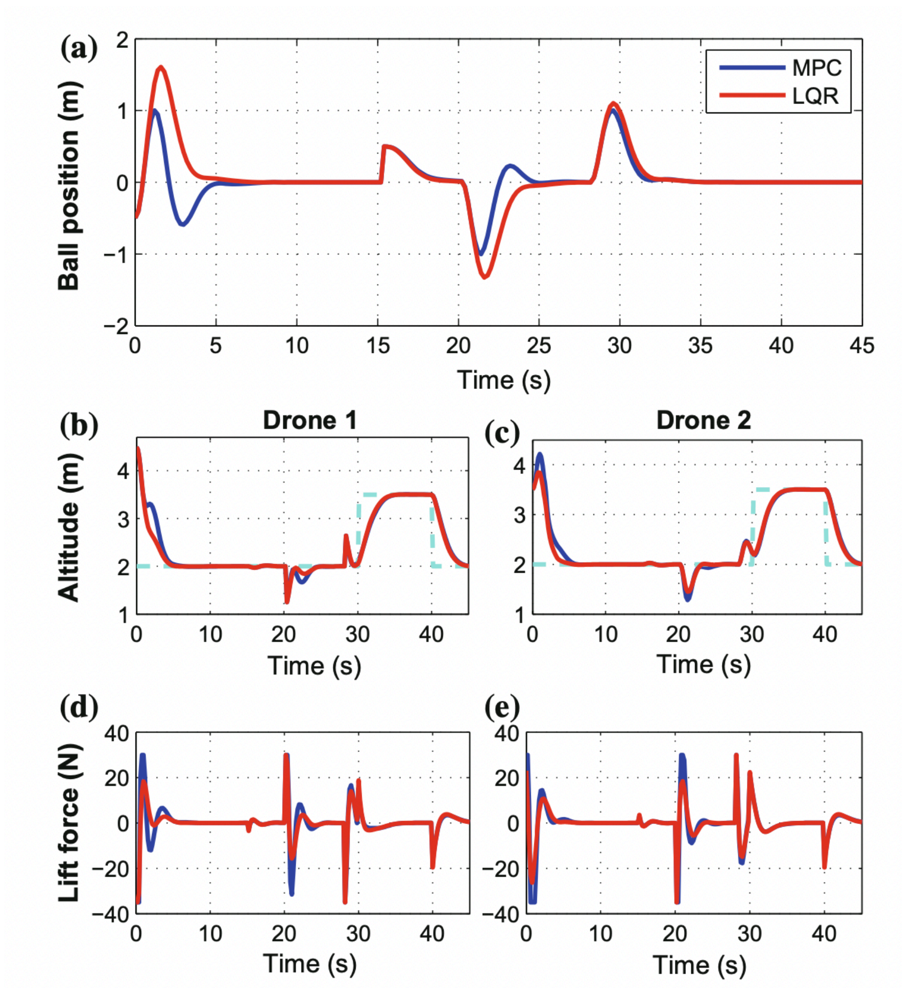
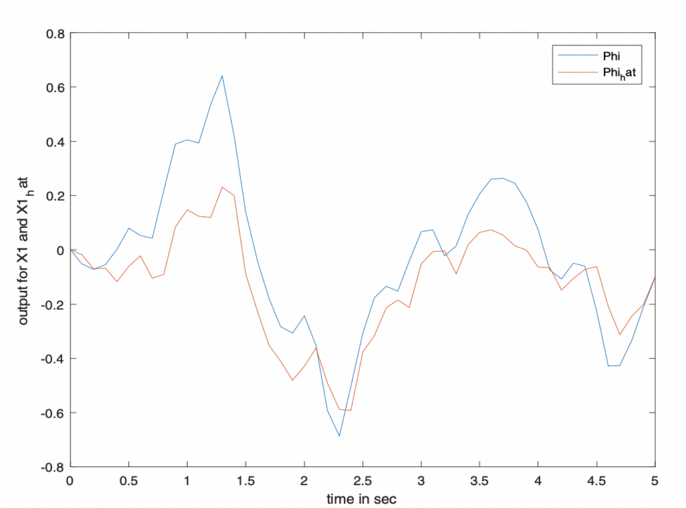


Figure (3) : Simulation result with disturbances. Credit to [1]

We have similar result to the previous one. LQR sometimes perform out of the bounded region. One thing to note that MPC didn't have an obvious delay with the disturbances case. We then can make a little conclusion for both controllers. For non-disturbances case, LQR runs faster than

MPC, but for disturbances case, the execution time is closed. However, there's an important fact that LQR sometimes violates the constraint, that is not allowed to happen for some applications. MPC, in contrast, respond to the goal slower but ensure to operate within the bounded region. There doesn't exist the so called best controller. It depends on what is the objective and how complicated are the constraints. If the constraints can't be violated, we should choose the MPC. If we emphasize on the execution time and sacrifice some constraints, LQR would be a better candidate.

The second paper review is **Linear Optimal Control of a Parrot AR Drone 2.0**. The main objective is to compare close loop control and close feedback loop control with a filter. LQR represents for the close loop feedback control, while Linear Quadratic Gaussian (LQG) represents for close loop feedback control with filter. The reason they chose to compare LQR and LQG is that LQR can't always have the whole states of the system, and we need a filter to estimate other states based on the states that are measurable. The filter here they applied was Kalman filter, which was gotten after computed by MATLAB code. The following figures are part of the simulation result for the LQG.



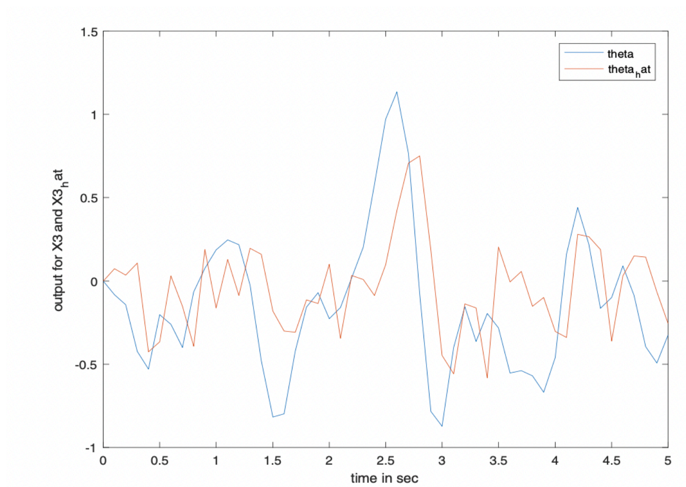
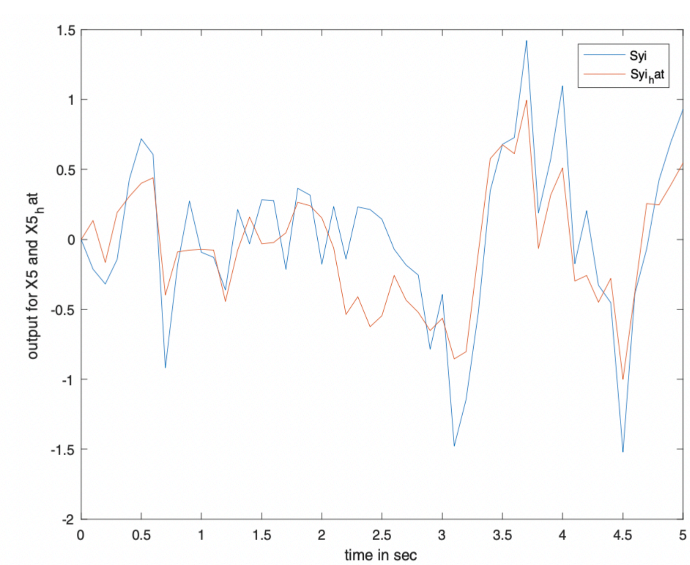


Figure (4) : Roll. Credit to [3]

figure (5) : Pitch. Credit to [3]



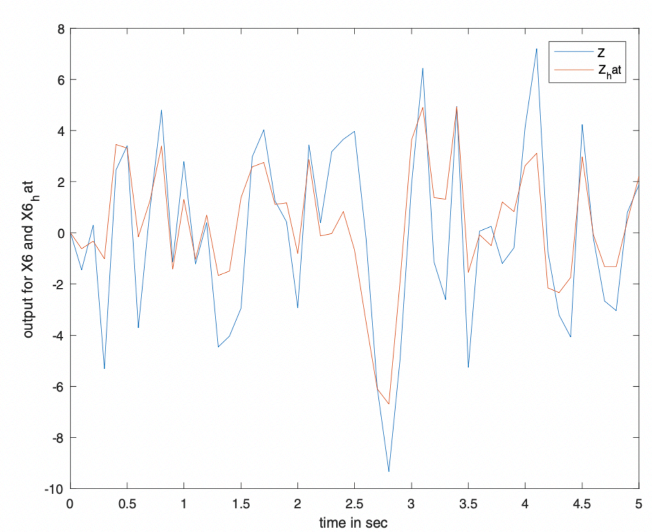


Figure (6) : Yaw. Credit to [3]

Figure (7) : Altitude Z. Credit to [3]

Since they didn't show every single figure of the parameters of LQR, I could only analyze some of them for LQG. As showing, the angle and the angle hat did have some errors, but it is in the allowable region. Also, the trend of the parameters is positively related. In my opinion, if I want do future research in this field, I'll probably use other filter or come up with ideas to turn a better Kalman filter.

Open Source Research

I found some open resources online that have built the model of drones, they provided the dynamics of a drone. There are slightly differences because of different purposes and scenarios. The basic dynamic of a single drone is to observe the position set (x, y, z) in 3D coordinates and the angle set (θ, ψ, ϕ), also we need their derivative to get velocity and angular velocity in different directions and angles. Then we can transform the dynamic into the common form for control problem.

$$x = \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ \theta \\ \psi \\ \phi \\ \dot{\theta} \\ \dot{\psi} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \end{bmatrix}, \quad \dot{x} = f(x, u), \quad u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}$$ The input u is the four motors.

Here, cited the dynamic scenario from [7] in figure (8) and (9).

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \ddot{\eta} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \frac{1}{m}(\cos(\psi)\sin(\theta)\cos(\phi) + \sin(\psi)\sin(\phi))U_1 + \frac{A_x}{m} \\ \frac{1}{m}(\sin(\psi)\sin(\theta)\cos(\phi) - \cos(\psi)\sin(\phi))U_1 + \frac{A_y}{m} \\ -g + \frac{1}{m}(\cos(\theta)\cos(\phi))U_1 + \frac{A_z}{m} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ M(\eta)^{-1}(\tau_\eta - C(\eta, \dot{\eta})\dot{\eta}) \end{bmatrix}$$

Figure (8) : The dynamic of a single drone. Credit to [7].

Variable	Definition	Variable	Definition
$\dot{x}, \dot{y}, \dot{z}$	Velocities in the body frame	U_1, U_2, U_3, U_4	Control inputs
$\ddot{x}, \ddot{y}, \ddot{z}$	Acceleration in the body frame	A_x, A_y, A_z	Aerodynamic disturbances
ϕ, θ, ψ	pitch, rolling and yaw- angle around the principle axes of the body frame	m	Mass of the quadrotor
$\dot{\phi}, \dot{\theta}, \dot{\psi}$	Angular velocities of the Quadrotor around the principle axes of the body frame	g	Gravitational constant
$\ddot{\eta}$	Angular accelerations of the Quadrotor around the principle axes of the body frame		

Figure (9) : Table of the variables. Credit to [7].

We then can use Matlab to solve the problem numerically. In Matlab, there's a function called lqr(A, B, Q, R). The input A, B are corresponding to our system dynamic $\dot{x} = Ax(t) + Bu(t)$, while Q is the state-cost weighted matrix and R is the input-cost weighted matrix, both depend on our objective. The output of the lqr in Matlab gives us a set (K, S, P). In this set, what we most concern should be the gain matrix K because using LQR, we need K to compute the feedback control u = - Kx to minimize the cost function $J(u) = \int_0^\infty (x^T Q x + u^T R u) dt$.

For LQG, Matlab also provides a function to generate control input u to regulate y to zero value, where y = Cx(t) + Du(t) + v. V is the disturbances.

Conclusion

After some literature reviews, There are still some methods to solve the optimal control problem for drones I haven't completely understood well such as dynamic programming (DP). In fact, the concept of DP was denoted decades ago. However, due to the computational efficiency and hardware limitations, DP was not applied commonly to solve optimal control problem. My next step could do research on other methods to solve the same problem, and compare pros and cons.

Reference

[1] Rubio, A.A., Seuret, A., Ariba, Y., Mannisi, A. (2016). Optimal Control Strategies for Load Carrying Drones. In: Seuret, A., Hetel, L., Daafouz, J., Johansson, K. (eds) Delays and

Networked Control Systems . Advances in Delays and Dynamics, vol 6. Springer, Cham. https://doi.org/10.1007/978-3-319-32372-5_11

[2] Timothy L. Molloy, Jairo Inga Charaja, Sören Hohmann, Tristan Perez, "Introduction", *Inverse Optimal Control and Inverse Noncooperative Dynamic Game Theory*, pp.1, 2022.

[3] S. Saha and S. Wadoo, "Linear optimal control of a parrot AR drone 2.0," *2017 IEEE MIT Undergraduate Research Technology Conference (URTC)*, 2017, pp. 1-5, doi: 10.1109/URTC.2017.8284178.

[4] K. Louadj, F. Demim, A. Nemra and P. Marthon, "An Optimal Control Problem of Unmanned Aerial Vehicle," *2018 5th International Conference on Control, Decision and Information Technologies (CoDIT)*, 2018, pp. 751-756, doi: 10.1109/CoDIT.2018.8394830

Open Source

[5] Rodrigues, H. S., Monteiro, M. T. T., & Torres, D. F. (2014). Optimal control and numerical software: an overview. *arXiv preprint arXiv:1401.7279*.

[6] Rutquist, P. E., & Edvall, M. M. (2010). Propt-matlab optimal control software. *Tomlab Optimization Inc*, *260*(1), 12.

[7] E.P. Biever, Controlling of a single drone, DC 2015.0xx, Eindhoven University of Technology, Dynamics and Control Group, Department of Mechanical Engineering, Eindhoven, The Netherlands, BSc Thesis, 2015.