Autonomous Flight Planning and Control

In this project, I was interested in applying model predictive control to a simulated quadrotor. The path planning and control were implemented in a known environment. This meant to simulate a sort of urban environment drone delivery task, although in that scenario it is likely there will be unknown obstacles as well.

Path Planning

The path planning in this project is done in a known environment with no need for replanning. Because of this, it isn't crucial that the algorithm execute extremely quickly; all of the planning is done before the quadcopter takes flight.

I decided to use an A* variant, θ^*, that essentially prunes unnecessary nodes as the path is built to result in a shorter, any angle path [1]. This is a simple planner that does not account for the dynamics of the vehicle. Instead a safety buffer is applied between obstacles and the path such that small deviations from the path do not result in collisions.

Equations of Motion

The nonlinear dynamic model of a quadrotor used in this project is derived by Subbarao [2]. The state of the system is:

$$X = [x, y, z, V_x, V_y, V_z, \phi, \theta, \psi, p, q, r]$$

Where p, q, r represent the angular velocities. The dynamics for the system are:

$$\dot{X} = \begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{V}}_{\mathbf{B}} \\ \dot{\Theta} \\ \dot{\Omega} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{BI}^T \mathbf{V}_{\mathbf{B}} \\ -\Omega \mathbf{V}_B + \mathbf{R}_{BI} \left(g \hat{\mathbf{e}}_{\mathbf{3}} \right) \\ \mathbf{W}(\phi,\, \theta,\, \psi)\Omega \\ \mathbf{J}^{-1} \left(-\Omega \times \mathbf{J}\Omega \right) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{1}{m} & -\frac{1}{m} & -\frac{1}{m} & -\frac{1}{m} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -J_x^{-1}L & 0 & J_x^{-1}L \\ J_y^{-1}L & 0 & -J_y^{-1}L & 0 \\ -J_z^{-1}c & J_z^{-1}c & -J_z^{-1}c & J_z^{-1}c \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix}$$

Subject to control:

$$U = [F_1, F_2, F_3, F_4]$$

See Subbaroo [2] for more details on the equations of motion for the system model.

Linear Model

Toward a discretized model of our dynamic system we can use Euler integration as follows:

$$X_{k+1} = f(X, U) \approx X_k + dt\dot{X}$$

We can linearize this model by performing a first order Taylor expansion of $f(X, U)$ about an operating point (X_o, U_o). This results in a linear model:

$X_{k+1} \approx c + A(X_k - X_o) + B(U_k - U_o)$

$c = X_o + dt * \dot{X}(X_o, U_o)$, $A = I + dt\frac{d\dot{X}}{dX}(X_o, U_o)$, $B = dt\frac{d\dot{X}}{dU}(X_o, U_o)$

MPC

In linear model predictive control, knowing the current state estimate, we predict the next N future states, each with a timestep dt, and find the optimal N - 1 control inputs to take at each step. We then execute the first control input and restart the process. That is:

Repeat:

1. Measure/estimate the state X_0. This will be our operating point
2. Linearize the model about the current state and control-- calculate c, A, X
3. Obtain the optimal control strategy for N steps of dt into the future
4. Apply the first element in the control strategy

In this implementation, the following were chosen:

$dt = 0.1, N = 20$

A LQR quadratic cost was chosen as the objective function for optimization with:

$Q = 5 * I$, $R = 0.3 * I$

Q_N was chosen to be the solution to the algebraic Riccati equation P:

$$P_{t-1} = Q + A^T P_t A - A^T P_t B (B^T P_t B + R)^{-1} B^T P_t A,$$

The controller sequentially tracked positional waypoints found as outputs of the path planning algorithm, updating the goal waypoint once the quadrotor had reached the proximity of the current goal until the final goal had been achieved.

Cited Resources

1. A. Nash, S. Koenig and C. Tovey. Lazy Theta*: Any-Angle Path Planning and Path Length Analysis in 3D. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2010.
2. Ru, P.; Subbarao, K. Nonlinear Model Predictive Control for Unmanned Aerial Vehicles. Aerospace 2017, 4, 31. https://doi.org/10.3390/aerospace4020031