Template of Report for PDM course

Group Names

***Abstract*— Keep short - the whole paper shall not exceed four pages plus references (Strict).**

I. INTRODUCTION

Objective: Navigation of a quadcopter in a dynamic environment with multiple obstacles moving in a random motion. Obstacles are represented by cylinders which may represent people moving around in an indoor environment.
Environment: We will use PyBulletDrone environment for this project
Assumptions: The instantaneous state of the system, including instantaneous positions and sizes of all obstacles, is available to the motion planning algorithm.
Procedure:

1) Start with a vanilla RRT^* algorithm that generates the trajectory in a static environment.
2) Sampling is done from a 12D C-space (Position, Velocity) taking the kinodynamics of the rotor into account.
3) Then make the quadcopter follow this path using an off the shelf control algorithm.
4) Implement different flavours of the RRT algorithm for dynamic environments.
5) Compare these algorithm on the basis of their compute time and the path length.

Questions: Should we search for the optimal path in the 3D World space or the 6D C-Space of the robot or 12D Position + Velocity space?

II. ROBOT MODEL

We have decided to use Quadrotor as our robot. Quadrotor is a type of unmanned aerial vehicle (UAV) or drone that is lifted and propelled by four rotors. Each rotor is mounted at the end of a horizontal arm extending from the central body of the vehicle.

Fig. 1: Quadrotor

III. WORKING OF QUADROTORS

The quadrotor is a highly non-linear, six degree-of-freedom and under-actuated system. It is under-actuated because inputs are only 4 from 4 motors and total degree-of-freedom is 6. It is controlled by varying the thrust forces of each rotor and balancing the drag torque. A quadrotor has two sets of counter-rotating propellers, therefore neutralizing the effective aerodynamic drag. It has four principal modes of operation(Figure 2):

1) Vertical movement is controlled by simultaneously increasing or decreasing the thrust of all rotors.
2) Yaw moment is created by proportionally varying the speeds of counter-rotating parts to have movement with respect to quadrotor's z-axis.
3) Roll can be controlled by applying differential thrust forces on opposite rotors of the quadrotor to have movement with respect to quadrotor's x-axis.
4) Pitch can be controlled by applying differential thrust forces on opposite rotors of the quadrotor to have movement with respect to quadrotor's y-axis.

IV. Quadrotor Model

A. Model of a rotor

Each rotor rotates with angular velocity ω and generates a lift force F and moment M. Moment is acting opposite to the directing of rotation. (Figure 3)

Fig. 3: Quadrotor arm

The lift Force F and moment M of ith rotor can be calculated by:

$F_i = k_f * \omega_i^2$ and $k_f = k_T * \rho * D^4$

$M_i = k_m * \omega_i^2$ and $k_m = k_Q * \rho * D^5$

where:
k_T is thrust coefficient
k_Q is torque
rho is fluid density
D is diameter of propeller

B. Equations of Motion

Fig. 4: Quadrotor in inertial and body frames

Total thrust and moment is the sum of individual ones in each of the 4 rotors.

Thrust: $F = F_1 + F_2 + F_3 + F_4 - mga_3$
Here, F_x are individual lift forces by the propellers and $m * g$ is the one by gravity.

Moment: $M = r_1 * F_1 + r_2 * F_2 + r_3 * F_3 + r_4 * F_4 + M_1 + M_2 + M_3 + M_4$
Here, $r_x * F_x$ are the moments created by forces in quadrotor's centre of gravity and M_x are the

individual moments created by the propellers.

C. State Variables

u, ϕ, p are the linear velocity along the roll-axis direction, angle rotated and the angular velocity about the roll-axis. w, ψ, r is the linear velocity along the yaw-axis direction, angle rotated and the angular velocity about the yaw-axis. v, θ, q is the linear velocity along the pitch-axis direction, angle rotated and the angular velocity about the pitch-axis. x, y, z are the position of quadrotor along a_1, a_2, a_3 axis of the inertial frame.

D. Newton-Euler Equations for Quadrotor

Linear Dynamics Applying Newton's Second Law for system of particles, we get (in inertial frame);

$F = mass * acceleration$

$acceleration(\ddot{r}) = d\dot{r}/dt$, where $\dot{r} = [u, v, w]^T$ (3.3)

Since, w is the yaw-axis in which we calculate thrust, we get;

$$mass{*}\ddot{r} = \begin{bmatrix} 0 \\ 0 \\ -m * g \end{bmatrix} + R_{\psi}\phi\theta \begin{bmatrix} 0 \\ 0 \\ F_1 + F_2 + F_3 + F_4 \end{bmatrix}$$

Rotational Dynamics Applying Euler's rotation equations, we get (in body frame);

$M_c =^A dH_c^B/dt =^B dH_c^B/dt +^A \omega^B \times H_c^B$

where, H_c is the angular momentum and ${}^A\omega^B$ is angular velocity of body B in frame A which is given by $p.b_1 + q.b_2 + r.b_3$

General vector form of Euler's equation is; $M_c = I\dot{\omega} + \omega \times (I\omega)$

For Quadrotor, after rearranging the general vector form;

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} L(F_2 - F_4) \\ L(F_3 - F_1) \\ M_1 - M_2 + M_3 - M_4 \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

Let $\gamma = k_M/k_F$, $M_i = \gamma F_i$, we get;

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} 0 & L & 0 & -L \\ -L & 0 & L & 0 \\ \gamma & -\gamma & \gamma & -\gamma \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

E. Final Joint Equation

Joint Equation using Linear and Rotational dynamics equations, we get;

$$\mathbf{u} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & L & 0 & -L \\ -L & 0 & L & 0 \\ \gamma & -\gamma & \gamma & -\gamma \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix}$$

where u is;

$$\begin{bmatrix} & thrust & \\ moment & about & x \\ moment & about & y \\ momemt & about & z \end{bmatrix}$$

V. Control Inputs

Control inputs are in the form of motor speeds for controlling thrust and 3 moments.

$$\begin{bmatrix} T \\ \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} k_F & k_F & k_F & k_F \\ 0 & Lk_F & 0 & -Lk_F \\ -Lk_F & 0 & Lk_F & 0 \\ k_M & -k_M & k_M & -k_M \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix}$$