Hybrid Cascaded MPC UAV System and Corresponding Waypoint Tracking Simulation

Xiaonan (Nice) Wang *

April 24, 2026

1 Mathematical Formulation of Cascaded Hybrid MPC System

We formulate cascaded hybrid UAV system as: a high-level Model Predictive Controller (MPC) acts as a kinematic trajectory generator to produce a continuously moving virtual target, which is subsequently tracked by a low-level PID controller.

1.1 Hybrid State Space

The overall system encompasses continuous physical states, the virtual target state from the MPC, and discrete logic states.

- **Continuous Physical States:** Position $p \in \mathbb{R}^3$, velocity $v \in \mathbb{R}^3$, orientation $\Theta \in \mathbb{R}^3$, and angular velocity $\omega \in \mathbb{R}^3$.
- **Continuous Virtual State:** Virtual target position $p_v \in \mathbb{R}^3$ driven by the MPC.
- **Discrete State:** The index of the current waypoint target $q \in \mathcal{Q} = \{0, 1, 2, ..., N\}$.

The complete hybrid state vector χ is defined as:

$$\chi = \begin{bmatrix} p \\ v \\ \Theta \\ \omega \\ p_v \\ q \end{bmatrix} \in \mathbb{R}^{15} \times \mathcal{Q} \qquad (1)$$

*wangxiaonannice@gmail.com

1.2 High-Level Kinematic Model Predictive Control (Discrete)

To prevent aggressive attitude oscillations, the MPC solves a finite-horizon Optimal Control Problem at discrete control intervals k to smoothly advance the virtual target p_v towards the absolute waypoint W_q.

The discrete-time linear prediction model for the virtual target is formulated as:

$$p_v(k+1) = Ap_v(k) + Bu_{mpc}(k) \tag{2}$$

where $A = I_{3\times3}$, $B = I_{3\times3}\Delta t$, and $u_{mpc} \in \mathbb{R}^3$ is the virtual velocity command.

At each step k, the MPC minimizes the following quadratic cost function over a prediction horizon H_p:

$$\min_{U_{mpc}} \sum_{i=0}^{H_p-1} \left(\|p_v(k+i) - W_q\|_Q^2 + \|u_{mpc}(k+i)\|_R^2 \right) \tag{3}$$

Subject to box constraints on the input velocity:

$$u_{min} \leq u_{mpc}(k+i) \leq u_{max}, \quad \forall i \in \{0, ..., H_p - 1\} \tag{4}$$

where Q and R are positive definite weight matrices penalizing tracking error and control effort, respectively. The first element of the optimal sequence $u^*_{mpc}(k)$ is applied to update the virtual target.

1.3 Low-Level Continuous Dynamics (Flow Map f)

Between discrete waypoint transitions, the system undergoes continuous-time physical evolution governed by the flow set C and flow map $f(\chi)$.

The flow set dictates that continuous flight is maintained as long as the UAV has not reached the tolerance radius ϵ of the current waypoint, or it is at the final waypoint:

$$C = \left\{ \chi \in \mathbb{R}^{15} \times \mathcal{Q} \;\middle|\; \|p - W_q\| \geq \epsilon \;\vee\; q = N \right\} \tag{5}$$

Within C, the continuous dynamics are decoupled into translational and rotational components to match the state dimensions:

$$\dot{\chi} = f(\chi) = \begin{bmatrix} \dot{p} \\ \dot{v} \\ \dot{\Theta} \\ \dot{\omega} \\ \dot{p}_v \\ \dot{q} \end{bmatrix} = \begin{bmatrix} v \\ \mathcal{F}_{trans}(p, v, \Theta, \omega, u_{pid}([p, v, \Theta, \omega], p_v)) \\ W(\Theta)\omega \\ \mathcal{F}_{rot}(\Theta, \omega, u_{pid}) \\ u^*_{mpc}(t) \\ 0 \end{bmatrix}, \quad \chi \in C \tag{6}$$

where $\mathcal{F}_{trans}$ represents the translational dynamics (acceleration bounded by thrust and gravity), and $\mathcal{F}_{rot}$ governs the rotational kinematics and attitude dynamics, both controlled by the PID loop.

It is worth noting that the evolution of the virtual target, denoted as $\dot{p}_v = u^*_{mpc}(t)$, is formulated in continuous-time (Newton's notation) to close the theoretical hybrid system flow map. In the actual digital simulation environment, this continuous derivative is realized through first-order Euler numerical integration at each control timestep Δt:

$$p_v(k+1) = p_v(k) + u^*_{mpc}(k)\Delta t \tag{7}$$

This discrete algebraic update seamlessly bridges the low-frequency optimal velocity commands generated by the discrete MPC solver with the high-frequency continuous physical tracking executed by the underlying PID controller.

1.4 Low-Leve Discrete Event Transitions (Jump Map g)

The discrete transitions (waypoint switching) are triggered strictly when the physical position p enters the ϵ-neighborhood of the current target W_q. The jump set D is defined as:

$$D = \left\{ \chi \in \mathbb{R}^{15} \times \mathcal{Q} \;\middle|\; \|p - W_q\| < \epsilon \;\wedge\; q < N \right\} \tag{8}$$

Upon intersecting D, the system state undergoes an instantaneous jump defined by the map $g(\chi)$. Physical states remain continuous, while the discrete target index advances:

$$\chi^+ = g(\chi) = \begin{bmatrix} p \\ v \\ \Theta \\ \omega \\ p_v \\ q+1 \end{bmatrix}, \quad \chi \in D \tag{9}$$

1.5 Overall Hybrid Automaton

Combining the flow and jump components, the overall cascaded system is written as the hybrid automaton $\mathcal{H}$:

$$\mathcal{H} : \begin{cases} \dot{\chi} = f(\chi), & \chi \in C \\ \chi^+ = g(\chi), & \chi \in D \end{cases} \tag{10}$$

2 Simulation: Waypoint Tracking

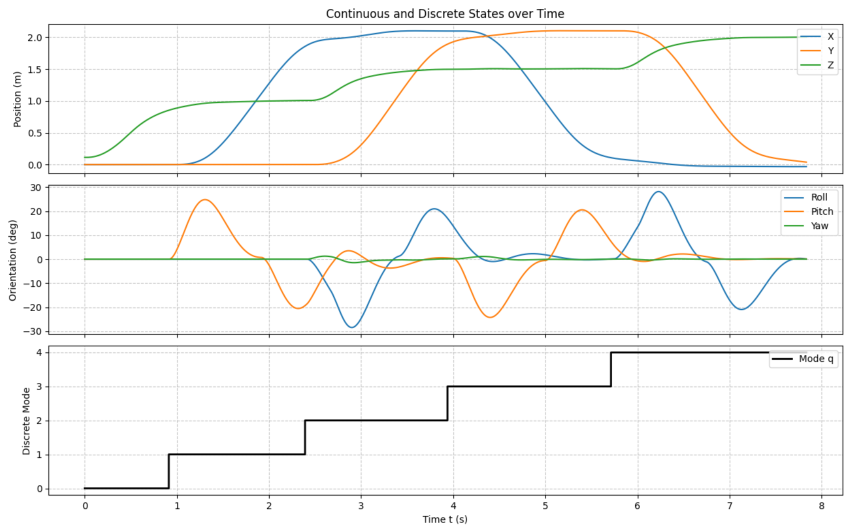

Figure 1: Waypoint Tracking Simulation: State Plot

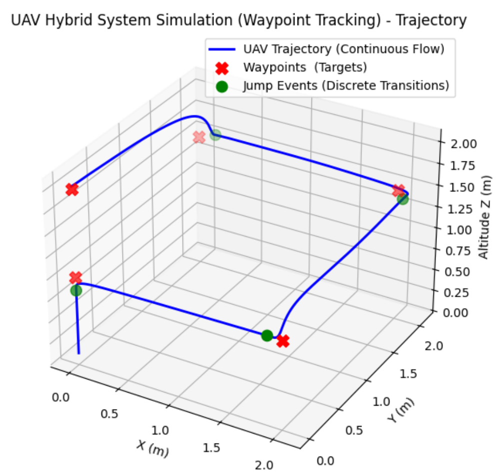

Figure 2: Waypoint Tracking Simulation: Trajectory Plot