Dubins Car Project: Comprehensive Summary

ECPS 208: Control Systems for Cyber Physical Systems

Spring 2025

Introduction

The Dubins Car project explored nonlinear control design for a simplified vehicle model, progressing from single-robot navigation to multi-robot coordination with obstacle avoidance. Our approach emphasized dynamic priority systems, robust safety via Higher-Order Control Barrier Functions (HOCBFs), and parameter tuning for reliability. Below, we summarize our accomplishments for each step, highlighting unique contributions.

Step 1: Modeling

- **Objective**: Derive the third-order nonlinear model.
- **Accomplishment**: Defined state vector $x = [p_x, p_y, \theta]^T$, with dynamics:

$$\dot{x} = \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ u_1 \end{bmatrix}$$

  where v is initially constant, and u_1 controls heading.
- **Unique Aspect**: Clarified nonholonomic constraints, providing a foundation for control design.

Step 2: Simple Control

- **Objective**: Steer to target (0,0) to (5,5).
- **Accomplishment**: Implemented proportional controller $u_1 = k \cdot \sin(\theta_d - \theta)$. Simulated navigation with constant velocity.
- **Result**: Reached target; identified need for a kill switch due to continuous motion.
- **Unique Aspect**: Tuned gain k, balancing responsiveness and stability, noting oscillations at high gains.

Step 3: Velocity Control

- **Objective**: Adjust velocity based on position.
- **Accomplishment**: Used distance-based control $v = \min(k_v \cdot d, v_{\max})$ in `step3.m` ($v_{\max} = 5$, $v_{\min} = 0.1$, $k_v = 1$).
- **Result**: Enabled precise stopping at the target.
- **Unique Aspect**: Compared heading- and distance-based methods, selecting the latter for effectiveness.

Step 4: Obstacle Avoidance

- **Objective**: Avoid a static obstacle using HOCBFs.
- **Accomplishment**: Defined safety function $h(x) = \|p - o\|^2 - r^2$, used HOCBFs for relative degree 2 in `step4.m` (obstacle at (5,5), radius 2).
- **Result**: Navigated to (10,8) while avoiding obstacle.
- **Unique Aspect**: Added dither and standstill mechanism for robustness.

Step 5: Two Robots Criss-crossing

- **Objective**: Coordinate two robots avoiding collisions.
- **Accomplishment**: Implemented HOCBFs and dynamic priority (closer robot gets weight 100) in `step5_2.m`. Initial positions: (1,2), (0,8); goals: (10,8), (10,2).
- **Result**: Collision-free paths, though large safe distances (4m) caused oscillations.
- **Unique Aspect**: Dynamic priority and safe distance analysis mitigated conflicts.

Step 6: Two Robots with Obstacle

- **Objective**: Add obstacle avoidance for two robots.
- **Accomplishment**: Integrated obstacle constraints in `step6.m` (obstacle at (5,5), radius 1.5, safe distance 3m).
- **Result**: Successful navigation to targets.
- **Unique Aspect**: Balanced inter-robot and obstacle constraints with priority rules.

Step 7: Three Robots Criss-crossing

- **Objective**: Extend to three robots with dynamic priority.
- **Accomplishment**: Modified `step7.m` for robots at (1,2), (0,8), (2,5) to (5,10), (12,3), (18,8). Used weights [100, 50, 1].
- **Result**: Efficient, collision-free paths.
- **Unique Aspect**: Dynamic priority reassignment ensured smooth coordination.

Step 8: Three Robots with Obstacle

- **Objective**: Include obstacle avoidance for three robots.
- **Accomplishment**: Enhanced `step8.m` with HOCBFs for obstacle (5,5, radius 1.5) and inter-robot safety (2m). Used QP fallback.
- **Result**: Robust navigation to goals.
- **Unique Aspect**: Fine-tuned parameters and implemented fallback for reliability.

Conclusion

Our project progressed from a single-robot model to a sophisticated three-robot system with obstacle avoidance. Key innovations included dynamic priority systems, HOCBF-based safety, and robust QP fallbacks, ensuring reliable navigation in complex scenarios.