Effectiveness of MPC Methods for Autonomous Tractor Trailer Parking

Avaniko Asokkumar
MEAM, Robotics
University of Pennsylvania
Philadelphia, Pennsylvania
aavaniko@seas.upenn.edu

Vanessa Gong
MEAM, Robotics
University of Pennsylvania
Philadelphia, Pennsylvania
van3ssa@seas.upenn.edu

Runing Guan
Robotics
University of Pennsylvania
Philadelphia, Pennsylvania
rnguan@seas.upenn.edu

***Abstract*—Reversing a tractor–trailer into a parking bay is a challenging control problem due to nonlinear and underactuated kinematics, nonholonomic constraints, and the inherent instability that arises during backward motion. This work evaluates the effectiveness of several model predictive control methods for autonomous trailer docking, using a combination of Hybrid A* planning, nonlinear trajectory optimization, and receding-horizon tracking controllers. A kinematic model of an off-axle truck–trailer system is used to generate dynamically feasible trajectories, and obstacle avoidance is incorporated through convex separation constraints. We implement three controllers—baseline MPC, an obstacle-aware MPC, and a hybrid switching strategy—and assess their robustness under disturbances including reduced friction, steering slip, and sensor noise. The findings demonstrate that while Hybrid A* and nonlinear trajectory optimization provide a strong foundation for generating feasible reverse-parking paths, the choice of controller plays a critical role in execution performance, with obstacle-aware formulations offering substantially improved robustness to disturbances compared to standard MPC.**

I. Introduction

A. The Problem

Reverse maneuvering of a truck–trailer system is difficult due to its nonlinear and underactuated kinematics, nonholonomic constraints, and the inherent instability that appears when reversing. Small deviations in steering or articulation can rapidly grow, causing the trailer to swing and potentially fold into a V-shaped configuration, commonly referred to as jackknifing. These characteristics make it challenging to position the vehicle accurately into a desired parking pose while avoiding collisions and staying within articulation and steering limits.

B. The Motivation

Truck trailers are widely used in logistics, agriculture, and transportation, where precise reverse maneuvers in constrained spaces are routine. Automating these maneuvers reduces operator workload, increases safety, and improves efficiency, especially in environments where jackknifing or path deviation can lead to costly delays or worse. Beyond practicality, the problem highlights fundamental challenges in planning and controlling unstable, coupled systems, motivating the need for robust modeling and trajectory generation approaches.

II. Methods

A. Kinematics Model

We model the vehicle using the General 1-Trailer (G1T) formulation presented in [1]. This model captures the nonlinear and coupled motion between the tractor and trailer through the articulation angle and the off-axle hitch geometry.

The system state is defined as

$$x = (x,\, y,\, \theta,\, \psi,\, \phi,\, v),$$

where (x, y) denote the tractor rear-axle position, θ is the tractor heading, ψ is the articulation angle between the tractor and trailer, ϕ is the steering angle, and v is the longitudinal speed.

The tractor motion follows the standard bicycle model:

$$\dot{x} = v\cos\theta, \qquad \dot{y} = v\sin\theta, \qquad \dot{\theta} = \frac{v}{L_1}\tan\phi.$$

The trailer articulation evolves according to the off-axle hitch dynamics:

$$\dot{\psi} = -\frac{v}{L_2}\sin\psi + \frac{v}{L_1}\left(\frac{M_1}{L_2}\cos\psi - 1\right)\tan\phi,$$

where L_1 is the tractor wheelbase, L_2 is the trailer drawbar length, and M_1 is the hitch offset between the tractor rear axle and the hitch point.

The control inputs are

$$u = (\dot{\phi},\, a),$$

representing steering rate and longitudinal acceleration. Steering angle and velocity are integrated forward in time, and the full kinematic model is enforced as equality constraints in the trajectory optimization.

B. Trajectory Optimization

We formulate the reverse parking problem as a nonlinear program (NLP) and initialize it using waypoints generated by a Hybrid A* planner, which supplies a kinematically feasible global path. The waypoints are upsampled via cubic-spline interpolation to create a smooth initial guess for both the state and input trajectories at the resolution required by the optimizer.

The NLP uses the full state trajectory $\{x_k\}$ and input trajectory $\{u_k\}$ as decision variables. Dual variables associated with obstacle-avoidance constraints are also included. Dynamic feasibility is enforced by imposing the G1T kinematic model as equality constraints at each point.

To model interactions with the environment, the truck, trailer, and obstacles are represented using half-space geometry following the formulation in [2]. The dual variables ensure convex and differentiable separation between vehicle geometry and obstacles, enabling reliable constraint handling within the solver. Operational limits are applied on steering angle ϕ, articulation angle ψ, trailer velocity v, and the control inputs a and $\dot{\phi}$.

A terminal constraint enforces that the final state x_N reaches the desired goal pose within a small tolerance. The cost function penalizes state-tracking error, control effort, and terminal error using a quadratic cost:

$$J = \sum_{k=0}^{N-1} (x_k - x_g)^\top Q (x_k - x_g) + u_k^\top R u_k + (x_N - x_g)^\top Q_f (x_N - x_g).$$

This encourages the trajectory to remain smooth, dynamically feasible, and tightly aligned with the desired parking configuration while satisfying all physical and geometric constraints of the articulated system.

C. Simulation Disturbances

To evaluate controller robustness, we introduce a realistic disturbance model that captures environmental effects and modeling uncertainties during closed-loop execution. The disturbances are applied at each simulation timestep and affect both the control inputs and the state evolution.

Reduced tire-road friction is modeled by attenuating the commanded acceleration. Given a friction coefficient $\mu \in [0, 1]$, the effective acceleration is $a_{\text{eff}} = \mu \cdot a$, where a is the nominal acceleration command from the controller. A value of $\mu = 1$ corresponds to ideal friction, while lower values (e.g., $\mu = 0.9$) simulate slippery surfaces where only a fraction of the commanded acceleration is achieved.

Tire slippage reduces the effectiveness of steering commands. Given a slippage coefficient $\eta \in [0, 1]$, the effective steering rate is $\dot{\phi}\text{eff} = \eta \cdot \dot{\phi}$, where $\dot{\phi}$ is the commanded steering rate. A value of $\eta = 1$ indicates no slippage, while $\eta = 0.8$ implies that 20% of the steering authority is lost due to tire slip.

To capture unmodeled dynamics, sensor noise, and environmental disturbances, additive Gaussian noise is applied to all state variables: $x_{k+1} = x_{k+1}^{\text{nom}} + w_k, \quad w_k \sim \mathcal{N}(0, \sigma^2 I_6)$, where σ is the standard deviation of the noise and I_6 is the 6×6 identity matrix. This noise is scaled by the simulation timestep dt and perturbs the position, orientation, articulation angle, steering angle, and velocity states in the MPC input.

D. Controller Formulation

To track the optimized parking trajectory and address disturbances during execution, we implement three receding-horizon controllers of increasing complexity. All controllers operate on the G1T kinematic model and solve a finite-horizon optimal control problem at each timestep.

The baseline MPC minimizes tracking error to the reference trajectory obtained from the trajectory optimization. This controller assumes that the reference path is collision-free and therefore includes only standard state and input constraints, such as bounds on steering angle, articulation angle, and velocity.

To account for disturbances that may push the vehicle toward obstacles, we extend the baseline controller to an Obstacle-Aware MPC (OAMPC). This controller augments the MPC problem with half-space collision constraints applied across the prediction horizon, allowing the optimization to reason explicitly about the geometry of the truck, trailer, and surrounding obstacles. The constraints are implemented using dual variables, which provide smooth, differentiable separation suitable for real-time optimization.

We additionally design a Hybrid Obstacle-Aware MPC that switches between the baseline MPC and the OAMPC depending on whether a collision is predicted within the MPC horizon. When the nominal trajectory remains safely clear of obstacles, the controller operates in the simpler baseline mode to reduce computation. When a potential collision is detected, it transitions to the obstacle-aware mode to enforce geometric safety. This hybrid strategy improves robustness to disturbances while avoiding unnecessary constraint handling in collision-free scenarios.

III. Results

A. NLP Optimization with Hybrid A* Initialization

We evaluate our trajectory pipeline across four parking scenarios with varying initial positions and orientations, and goal positions. In each case, the Hybrid A* planner provides a kinematically feasible initial guess (green), and the NLP refines this path into a trajectory that satisfies the G1T kinematics, enforces obstacle constraints, and minimizes a cost that balances tracking accuracy and control effort. The optimized trajectories are shown in red, and the executed simulation traces are plotted in blue.

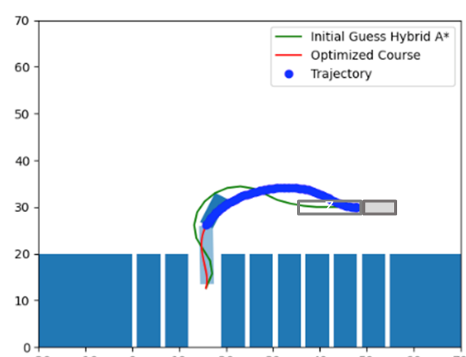

Fig. 1. Trajectory for Test Case 1

Test 1: Perpendicular Right Offset. Starting from $[50, 30, \pi/2, 0, 0, 0]$, this test evaluates the fundamental ability to correct a lateral offset while aligning the trailer perpendicularly to the parking slot.

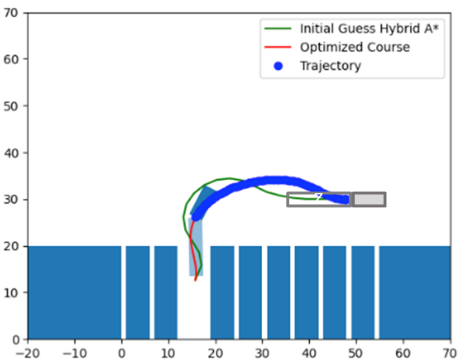

Fig. 2. Trajectory for Test Case 2

Test 2: Downward Right Offset. Starting from $[70, 60, 0, 0, 0, 0]$, this test examines performance when the vehicle approaches the parking spot with a forward facing orientation that requires a large heading change.

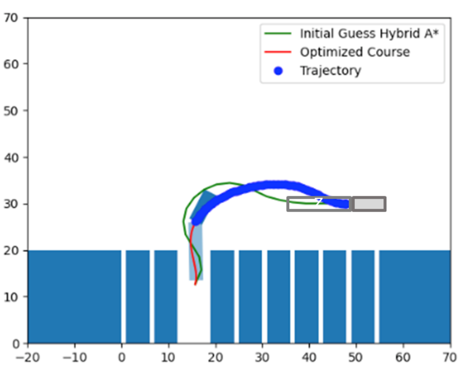

Fig. 3. Trajectory for Test Case 3

Test 3: Perpendicular High Offset. Starting from $[12, 60, \pi/2, 0, 0, 0]$, this test evaluates the system's ability to handle a large vertical distance before reaching the slot.

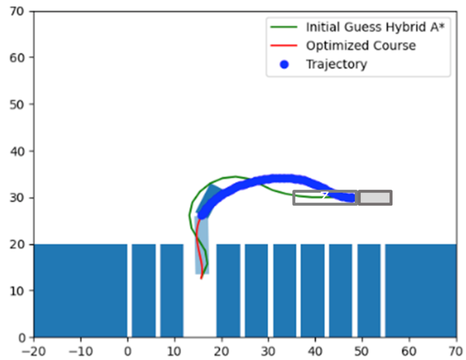

Fig. 4. Trajectory for Test Case 4

Test 4: Perpendicular Low Offset. Starting from $[12, 23.5, \pi/2, 0, 0, 0]$, in contrast to Test 3, this test assesses performance under tight proximity to the parking area, representative of narrow parking corridors.

Across all four tests, the NLP consistently reduces curvature and produces smoother trajectories than the Hybrid A* initialization. The results show the importance of combining a kinematic planner with a dynamics-aware optimization.

B. Controller Performance Under Disturbances

To evaluate the robustness of the controller formulations, we measure their performance on Test Case 1 under two conditions: (i) no external disturbances and (ii) random disturbances applied during execution. The success rate is defined as the percentage of simulation rollouts in which the vehicle reaches the goal pose without colliding with any obstacles. Table I reports performance in the disturbance-free setting, while Table II summarizes the results when disturbances are introduced.

TABLE I
TEST CASE 1 METRICS UNDER NO DISTURBANCES

Metric	MPC	OAMPC	Hybrid
LQR Score Avg	304.20	353.00	4750.60
Success Rate (%)	0	100	100
Obstacle Collision	Y	N	N
Min Solve Time(ms)	5.80	63.651	20.037
Max Solve Time(ms)	19.26	7511.62	2999.36
Average Solve Time(ms)	7.16	269.24	327.61
Distance (m)	0.31	5.37	3.07
Tractor Heading (deg)	-0.21	23.63	19.13
Hitch Angle (deg)	8.69	-32.32	-4.70

Both OAMPC and Hybrid OAMPC controllers had much longer solve times than the standard MPC one, but remained collision-free on solvable test-cases, and in scenarios where disturbances were favorable. The standard OAMPC NLP solver, while only using inactive collision constraints, takes longer on average to solve, due to the extra set of constraints involved, with 8 half-spaces generating 8 inequalities at each horizon timestep.

TABLE II
TEST CASE 1 METRICS UNDER RANDOM DISTURBANCES

Metric	MPC	OAMPC	Hybrid
LQR Score Avg	1248.18	1369.13	2836.38
Success Rate (%)	60	80	60
Obstacle Collision	Y	N	N
Min Solve Time(ms)	5.83	56.36	6.00
Max Solve Time(ms)	26.68	1633.75	5974.44
Average Solve Time(ms)	7.13	116.98	130.76
Distance (m)	0.29	1.28	0.72
Tractor Heading (deg)	1.26	5.27	3.87
Hitch Angle (deg)	-8.99	-4.78	-10.79

IV. DISCUSSION AND FUTURE WORK

A. *Reflection*

In conclusion, our combined Hybrid A* and nonlinear trajectory optimization pipeline proved effective for generating smooth and dynamically feasible parking paths. The NLP reliably refined the geometric Hybrid A* paths, and the obstacle-aware MPC variants performed as expected. The obstacle-aware variants had higher computational overhead, but largely avoided obstacles. In cases where the NLP solver was unable to solve (due to hard enforcement of collision constraints) the controller would input nominal values.

Not everything worked as expected. We encountered sensitivity to initialization quality and computation time, especially when obstacle constraints were active. Despite these challenges, the project demonstrated the value of pairing a kinematically aware planner with a dynamics-based optimizer. If compute were scaled, there is potential for even the obstacle aware MPC variants to run and solve in real-time.

Earlier in the project, we also attempted to use an RRT-based planner, but this approach was ultimately unsuccessful. The primary issue was that the RRT planner did not incorporate the G1T kinematic model or any dynamics. As a result, it produced forward-only paths that were completely incompatible with tractor–trailer motions. Because the RRT solutions did not satisfy the constraints, between the tractor and trailer, the NLP could not refine these paths into valid trajectories.

We also tried a fuzzy logic–based MPC model early in the project, but this approach was abandoned. The controller struggled to generate consistent steering and articulation corrections, and its rule base became increasingly unstable as the parking scenario grew more complex. As a result, it could not reliably guide the system toward the parking goal, especially in cases requiring precise trailer alignment or large heading reversals.

B. *Future Work*

Improving computation time is a key step toward real-time deployment. Converting the obstacle-aware MPC to a fast QP-based formulation could significantly reduce solve times and improve responsiveness.

Another direction is enabling online re-planning for environments with moving or uncertain obstacles. A short-horizon local planner could refresh a feasible path at each control cycle, warm-starting the MPC and providing a fallback policy when the optimization becomes temporarily infeasible.

Camera-assisted docking is another potential extension. Rear-facing cameras with visual markers or AprilTags placed at the bay could refine pose estimation near the goal, reducing map errors during the final backing maneuver.

Finally, online map validation would enhance safety by checking whether observed features match the expected environment. Detecting a map mismatch would allow the system to re-plan or abort before committing to a trajectory, preventing unsafe maneuvers.

References

[1] J. Lukassek, S. Danzer, B. Alrifaee, and O. Stursberg, “Kinematic modeling and analysis of general one-trailer systems with off-axle hitching,” IEEE Trans. Intell. Veh., 2024.

[2] A. J. Barry, A. Majumdar, and R. Tedrake, “A convex optimization approach to smooth and efficient vehicle manoeuvres with obstacle avoidance,” in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2018.

[3] MCastelyns, “BEP-Motion-planning-for-Truck-Trailers,” GitHub repository, 2025. [Online]. Available: https://github.com/MCastelyns/BEP-Motion-planning-for-Truck-Trailers. [Accessed: Nov. 09, 2025]