Minimal Lap Time Optimization-Based Motion Planning Considering Pit Stops

Hansung Kim *3037384309*
Chris Li *3037424116*
Daniel Lin *3036341060*
Youngsang Suh *3037437845*
Enpei Zhao *3037391148*

***Abstract*—In this work, we propose a formulation of an optimal race line trajectory generation problem considering pit stops. By adding new state variables such as fuel level and tire health index (from 0 to 1) as well as two binary variables in making decisions to make a pit stop for renewal (updated to 1) of the aforementioned new state variables. Existing approaches for minimal time trajectory optimization method that utilizes complex vehicle dynamics and tire forces was adopted and expanded to formulate a MINLP solved using Bonmin on CasADi. The optimal trajectory generated by the path planner module is then fed into an MPC path following module. The acquired solution demonstrates that given the assumption that there are no model uncertainties and full state is available, the time optimal trajectory planner is able to find a solution with making a pit stop after lap 1 to refuel the objective to complete 2 laps of a custom map. While path tracking resulted in a sub-optimal performance, possible solutions to enhance the path tracking performance as well as the robustness of the optimal trajectory planner is further discussed. Presentation video link**

I. Introduction

In Formula 1 races, cars endure high mechanical and thermal loads during the race which usually takes 90 to 180 minutes. Thus, pit stops for maintenance work such as wing repairs, tire changes, and refueling during the race are inevitable and critical for maximizing performance, which present an interesting optimization problem.

While many researchers have solved optimal trajectory planning problem for a single autonomous race car in various approaches, as shown in [1] [2] [3], most of them do not consider the nature of limited resources and necessity of pit stops in racing scenarios or assume non-degrading performance and unlimited resources. In solving this optimal control problem, a hierarchical controller architecture with trajectory optimization and model predictive controller to track the optimal trajectory is effective and attractive given its ability to consider constraints (track boundaries and dynamic boundary conditions) and make predictions about the car's states including the fuel level and tire health.

Our approach is mainly inspired by [1] and [4] and will adopt the same dynamic double track model and optimization approach with additional decision variables and models to consider limited resources constraints and simulate pit stops.

II. Optimal Time Trajectory Planner

A. Models

1) Track model: A simple test track with a constant track width of 4 meters (to the left and right) was created. The map is represented as .csv files with four parameters including 2D Cartesian coordinates of the centerline of the track and left and right track width at a given point. In order to smooth out the reference line noise introduced during map discretization, these imported maps were pre-processed in four steps: 1) linear interpolation to a small step size. 2) spline approximation for noise removal 3) track width correction due to the slight discrepancy from spline approximation 4) spline interpolation to the desired step size [1]. These steps improve the smoothness of the curvature and create a uniform step size along the reference line that can be used by the optimal lap time trajectory planner [1].

2) Vehicle model: It is appropriate to use a high fidelity vehicle dynamic model for high speed and large slip motion such as Formula 1 races [5]. To consider tire non-linearity, normal load dependent tire model, and describe load transfers between the wheels, a double track model—as shown in Fig. 1—was used for optimal lap time trajectory generation which are given by:

$$\begin{aligned}\dot{v} = \frac{1}{m}[&(F_{x,rl})cos(\beta) + (F_{x,fl}+F_{x,fr})cos(\delta-\beta)\\ &+(F_{y,rl}+F_{y,rr})sin(\beta) - (F_{y,fl}+F_{y,fr})sin(\delta-\beta)\\ &-\frac{1}{2}c_d\rho A v^2 sin(\beta)]\end{aligned} \quad (1)$$

$$\begin{aligned}\dot{\beta} = -\omega_z + \frac{1}{mv}(&-(F_{y,rl}+F_{x,rl})sin(\beta)\\ &+(F_{x,fl}+F_{x,fr})sin(\delta-\beta) + (F_{y,rl}+F_{y,rr})cos(\beta)\\ &+(F_{y,fl}+F_{y,fr})cos(\delta-\beta) + \frac{1}{2}c_d\rho A v^2 sin(\beta))\end{aligned} \quad (2)$$

$$\begin{aligned}\dot{\omega_z} = \frac{1}{J_{zz}}[&(F_{x,rr}-F_{x,rl})\frac{tw_r}{2} - (F_{y,rl}+F_{y,rr})l_r\\ &+((F_{x,fr}-F_{x,fl})cos(\delta) + (F_{y,fl}-F_{x,fr})sin(\delta))l_f]\end{aligned} \quad (3)$$

where equation (1) and (2) describe the longitudinal and lateral momentum with respect to the CoG [4]. Equation (3) describes the yaw motion of the vehicle. In this model, the state variables are v (the velocity at the center of gravity), β (the side slip angle), and ω_z (yaw rate). The vehicle parameters are the vehicle mass m, width w, the mass moment of inertia with respect to the vertical axis J_{zz}, the wheelbase length l, the distances between CoG and the front and rear axle l_f and l_r. Further, tw_f and tw_r are the track widths at the front

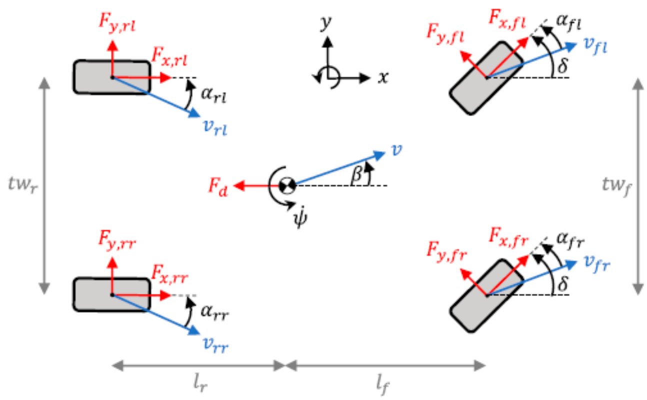

Fig. 1. Double track model [4]

and rear axle respectively. The $F_{y,ij}$ and $F_{x,ij}$ terms refer to the longitudinal and lateral tire forces, respectively. For more detailed explanation of the double track model refer to [4].

B. Problem Formulation

Our minimal time trajectory optimization problem adopts the same formulation as [4] with additional binary decision variables γ_{fuel} and γ_{tire}, decision variable to make a pit stop to refuel or change tires. Furthermore, additional constraints are formulated to model the fuel and tire consumption/degradation model (Eq. (8f) and (8h)) as well as refueling and tire changes as shown in Eq. (8j) and (8k). In addition to the double track model for vehicle dynamics, road and vehicle tracking model using a curvilinear coordinate system from [4] is adopted. Here, ξ is the relative angle to tangent on reference line and n is the lateral displacement from the reference line. Also, κ is the curvature of the reference line

$$\dot{s} = \frac{vcos(\xi + \beta)}{1 - n\kappa} \tag{4}$$

$$\dot{n} = vsin(\xi + \beta) \tag{5}$$

$$\dot{\xi} = \omega_z - \kappa \frac{vcos(\xi + \beta)}{1 - n\kappa}. \tag{6}$$

Time can be transformed into a dependent variable by using the path coordinate, s as an independent variable [4]. One advantage of this approach is that since s becomes an independent variable, the curvature doesn't change based on the inputs. It is especially meaningful in the cases that we need accurate guess of the solution for instance, when solving time dependent formulations [2]. The transformation takes the following form:

$$SF = \frac{dt}{ds} = \frac{1 - n\kappa}{vcos(\xi + \beta)} \tag{7}$$

In formulating the cost function for minimal time, the total elapsed time $\int_{t_0}^{t_f} dt$ can be transformed in terms of s as shown in the following cost function:

$$\begin{aligned} \min \quad & \int_{s_0}^{s_f} SF\, ds \\ & + r_\delta [\delta_0, ..., \delta_{N-1}] D^T D [\delta_0, ..., \delta_{N-1}]^T \\ & + r_F [F_0, ..., F_{N-1}] D^T D [F_0, ..., F_{N-1}]^T \end{aligned}$$

$$\text{s.t.} \quad x_{k+1} = f_d^c(x_k, u_k) \tag{8a}$$

$$g(x_k, u_k) = 0 \tag{8b}$$

$$h(x_k, u_k) \leq 0 \tag{8c}$$

$$x_k \in \mathcal{X}_{track} \tag{8d}$$

$$u_k \in \mathcal{U} \tag{8e}$$

$$fuel_{k+1} = fuel_k - \frac{1}{510} * \Delta s \tag{8f}$$

$$0 < fuel_k \leq 1 \tag{8g}$$

$$tire_{k+1} = tire_k - \frac{1}{1000} * \Delta s \tag{8h}$$

$$0 < tire_k \leq 1 \tag{8i}$$

$$fuel_k = fuel_k + \gamma_{fuel} * (1 - fuel_k) \tag{8j}$$

$$tire_k = tire_k + \gamma_{tire} * (1 - tire_k) \tag{8k}$$

$$\gamma_{fuel} \in \{0, 1\}, \ \gamma_{tire} \in \{0, 1\}. \tag{8l}$$

In order to prevent possible oscillations in solutions and assure convergence, regularization terms are added in the cost function [4]. According to [4], adding the regularization terms can smooth the progression of the control variables (δ_k and F_k). For detailed derivation and explanation of the cost function, refer to [4]. By introducing binary decision variables, the problem becomes a MINLP which was solved using the Bonmin solver on the CasADi interface. We have added new state variables, $fuel_k$ and $tire_k$ which indicate the current fuel level and tire health index (from 0 to 1), respectively. Also, arbitrary dynamic equations were added to model fuel consumption and tire degradation. Given new state variables and constraints is formulated such that the optimization solver will consider the fuel level and tire health index in making a decision to make a pit stop by updating the binary decision variables: γ_{fuel} and γ_{tire}. As singularity of s variable occurs when $v = 0$, it is difficult to simulate a pit stop dynamically. Thus, pit stop is simulated by adding a constant per γ_{fuel} or γ_{tire}, which is set to 5 seconds for our problem, to the lap time.

III. MPC Path Follower

After the optimal lap time trajectory is generated by the high-level planner, the reference trajectory is fed into the path follower MPC controller. For real-time control implementation, a low-fidelity kinematic bicycle model with two control inputs (acceleration and steering angle) was used instead of the high-fidelity double track model used for lap time optimization (the consequences of this design choice will be discussed later). The details of the kinematic bicycle model is omitted as it is a well-known model.

Although the optimization problem for optimal lap time trajectory generation was formulated with respect to space, s and in curvilinear coordinate system, it can output the optimal trajectory as time series of the four states used in the kinematic bicycle model: x, y, ψ, and v. The reference trajectory was interpolated to fit a uniform time interval, $T_s = 0.2$ seconds. Then we iteratively ran the MPC controller T times, where T is the length of the total interpolated reference trajectory. Assuming the full states of the vehicle are available with zero measurement error and no modeling uncertainty, a simple reference tracking MPC can be formulated to follow the optimal trajectory:

$$\min_{z,u} \quad \sum_{k=0}^{N} (z_k - \bar{z}_k)^T Q (z_k - \bar{z}_k) + \sum_{k=0}^{N-2} (u_{k+1} - u_k)^T R (u_{k+1} - u_k)$$

subject to

$$z_{k+1} = f_c^d(z_k, u_k),\ k = 0, 1, ..., N-1 \tag{9}$$

$$z_k \in \mathcal{Z}_{track},\ k = 0, 1, ..., N, \tag{10}$$

$$u_k \in \mathcal{U},\ k = 0, 1, ..., N-1, \tag{11}$$

$$z_0 = \mathbf{0}, \tag{12}$$

where z_k, $\bar{z}_k$, and u_k are the state of the vehicle, reference state from the optimal trajectory, and control input at time step k, respectively. The discrete time vehicle dynamics equations are acquired using the forward Euler discretization method with a sampling time of $T_s = 0.2$ seconds. The quadratic cost includes the reference tracking terms and penalizes jerk and the derivative of the steering angle for smoother tracking. Note that the vehicle dynamics is nonlinear, thus this optimization problem is a NLP. It was solved using the IPOPT solver on the CasADi interface.

IV. Results

A. Optimal lap time trajectory

Some of the key parameters used for the vehicle model and the dynamic constraints are listed in Table I. Furthermore, we set the total number of laps to two as any higher number of laps were computationally intractable.

Adding binary variables for every discretized step increases the number of variables to optimize. Unfortunately, due to the curse of dimensionality, as number of variables increases, an optimization problem becomes intractable especially when there's no possible analytic approach for solutions. Similarly, for this problem, solving for more than 3 laps make it impossible to solve. Here, when the solver realize the pit stop is required in i-th lap, it needs to get back to initial state and run optimization process entirely again, even matching every fuel or tire state to the previous iteration for continuity. Thus, although given constraints make a feasible set and it's quite simple for humans to figure out whether the vehicle would

TABLE I
VEHICLE PARAMETERS AND DYNAMIC CONSTRAINTS USED FOR OPTIMIZATION

Parameter	Value
m [kg]	1200
l_f [m]	1.6
l_r [m]	1.4
w [m]	2
$\delta_{s,min}$ [rad]	-0.35
$\delta_{s,min}$ [rad]	0.35
v_{min} [m/s]	1
v_{max} [m/s]	70
β_{min} [rad]	$-\frac{\pi}{2}$
β_{max} [rad]	$\frac{\pi}{2}$
$\omega_{z,min}$ [rad/s]	$-\frac{\pi}{2}$
$\omega_{z,max}$ [rad/s]	$\frac{\pi}{2}$

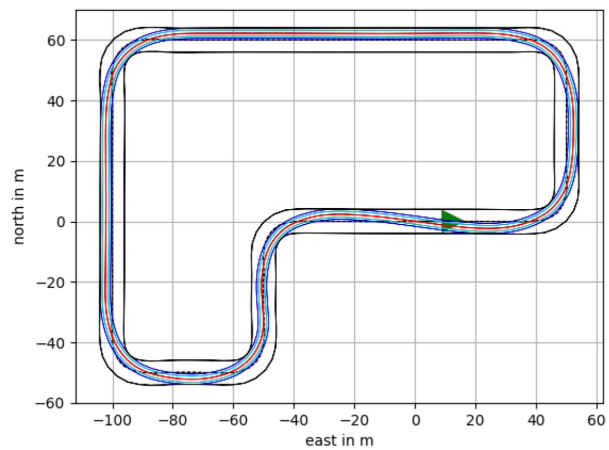

Fig. 2. Optimal lap time trajectory

need a pit stop or not, Bonmin solver cannot easily return feasible solutions. We were only able to compute a solution up to maximum 2 laps.

We arbitrarily assumed fuel/tire degradation constants to impose the vehicle to make a pit stop in every 1 and 2 laps, respectively. Thus, setting the total number of laps to two should return a solution that we should make a pit stop for fuel recovery only. Changing tires would not be an optimal solution since tire is still healthy for running two laps. According to the solution, we successfully retrieved the solution $\gamma_{tire} = 1$, $\gamma_{tire} = 0$.

The optimal lap time trajectory with a pit stop after the first lap is shown in Fig. 2. The red trajectory is the optimal lap time trajectory and the dark blue lines represent the boundary given the vehicle width to ensure that the vehicle stays inside the track at all times.

Another essential component of the optimal time trajectory is the velocity profile. The velocity profile can be visualized as shown in Fig. 3. Green and red regions represent sections of the trajectory where the vehicle is accelerating and decelerating, respectively.

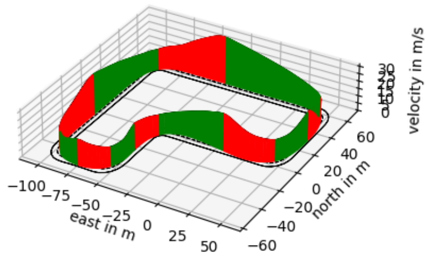

Fig. 3. Optimal lap time trajectory with velocity profile

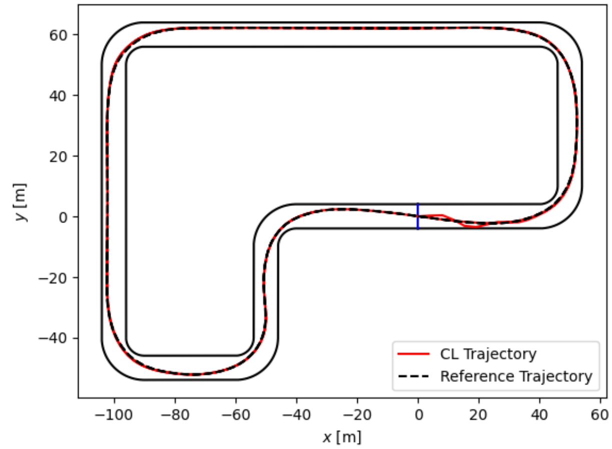

Fig. 4. Reference tracking

B. Path following

After some tuning, identity matrices were chosen for the cost matrices Q and R. The reference tracking results are shown in Fig. 4. Note that there are some oscillatory swerving about the reference trajectory in the beginning of the first lap. While the exact cause of this behavior is unidentified, some correlation with the initial states of the vehicle for the path following simulation and the erratic behavior has been observed. Otherwise, the MPC path follower is able to control the vehicle to follow the x and y trajectories with maximum deviation of 6.8 m and 5.8 m, respectively, as shown in Fig. 5. For ψ and v tracking, maximum tracking error was 0.5 radians and 2.2 m/s, respectively as shown in Fig. 5, 6, and 7. As previously discussed, fluctuation of ψ was observed in the beginning of the trajectory. Also, a spike in ψ tracking error occurs again around $t = 15$ seconds when the vehicle is entering the 4th corner on the track. Due to the tracking error, the actual race completion time using this MPC path follower is higher than the optimized lap time by 4.6 seconds. This value is an upper bound estimation by dividing the distance from the finish line to the current location of the vehicle at t_{opt} or when the optimal lap time is reached by the terminal velocity of the vehicle at t_{opt}. Please find the simulation demonstration video for the path following MPC here: video

The tracking performance can be improved by using a higher fidelity vehicle dynamics model. Although it is computationally attractive, the kinematic bicycle model may be inappropriate to follow the reference trajectory which was optimized while considering complex physics of tire forces, road friction, and tire load transfer. Without relaxing the dynamic constraints on the MPC path follower, its tracking performance is limited. Alternatively, instead of tracking the Cartesian representation of the state variables, MPC path follower can be reformulated with curvilinear representation of the state variables while still using the time domain model [2]. This technique formulates the cost function such that it maximizes the progress (s) over the horizon for a fixed sampling time. Using this technique, interpolation is no longer needed which could further improve the tracking performance.

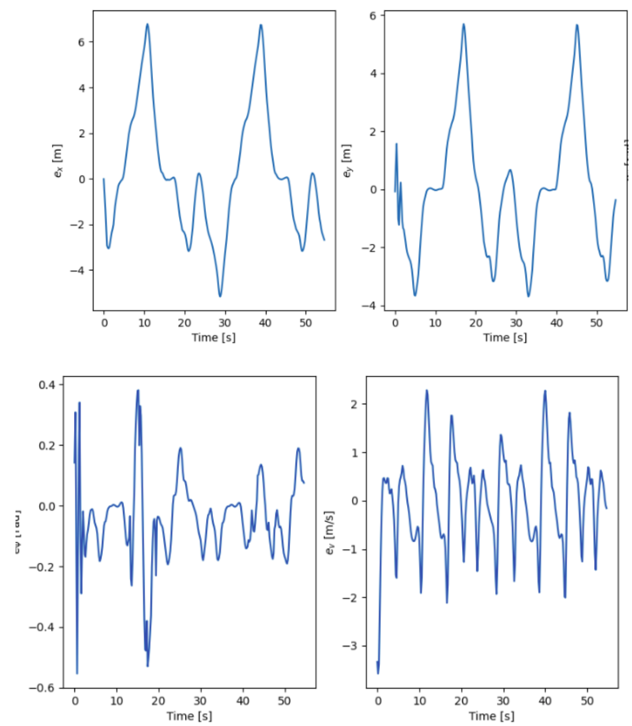

Fig. 5. Tracking errors for each state variables

V. Conclusion

In this project, we have formulated a minimal time racing problem with consideration of additional state variables that represent limited resources. Fuel consumption and tire degradation was arbitrarily modeled and pit stops were simulated mathematically but not dynamically due to a singularity issue as previously discussed in section II. Although we were able

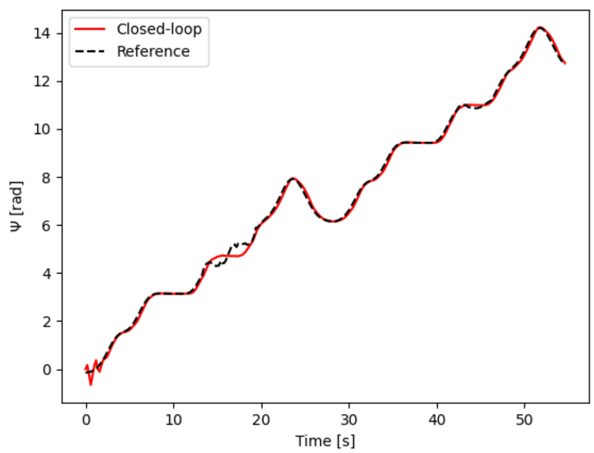

Fig. 6. ψ angle tracking time trace

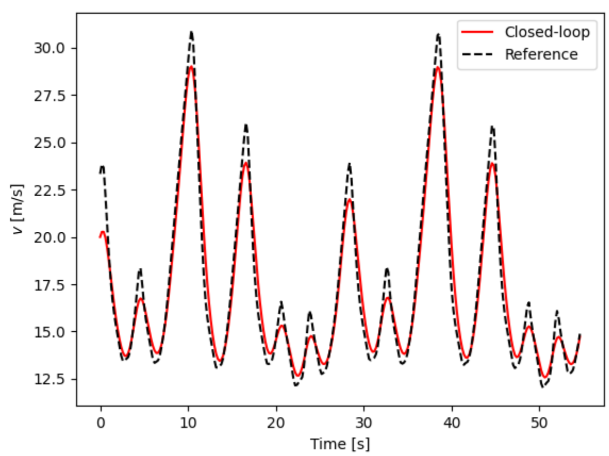

Fig. 7. velocity tracking time trace

to solve this optimization problem for a small total number of laps, it can be further improved. Reducing the number of variables would be the fundamental solution. Perhaps, maintaining one binary variables for fuel/tire state per lap can be the simplest method. Another possible method, we may try a different way to compute spline interpolation. Gauss-Legendre collocation that we implemented in this project to compute spline estimation increases the number of variables to optimize since it needs several points to guess the closest spline. Thus, finding a computationally efficient method to return spline interpolation would also be a reasonable future work. Furthermore, path tracking using a simple reference tracking MPC resulted in suboptimal performance. It could be replaced by a curvilinear MPC for enhanced tracking performance, which provides a new approach in minimal time reference tracking that wasn't possible in Cartesian coordinates.

Acknowledgment

The foundation of our optimal lap time trajectory generation code was based on the existing work done by the members of the TUMFTM.

References

[1] A. Heilmeier, A. Wischnewski, L. Hermansdorfer, J. Betz, M. Lienkamp, and B. Lohmann, "Minimum curvature trajectory planning and control for an autonomous race car," Vehicle System Dynamics, vol. 58, no. 10, pp. 1497–1527, 2019.

[2] J. L. Vazquez, M. Bruhlmeier, A. Liniger, A. Rupenyan, and J. Lygeros, "Optimization-based hierarchical motion planning for Autonomous Racing," 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2020.

[3] A. Rucco, G. Notarstefano, and J. Hauser, "An efficient minimum-time trajectory generation strategy for two-track Car Vehicles," IEEE Transactions on Control Systems Technology, vol. 23, no. 4, pp. 1505–1519, 2015.

[4] F. Christ, A. Wischnewski, A. Heilmeier, and B. Lohmann, "Time-optimal trajectory planning for a race car considering variable tyre-road friction coefficients," Vehicle System Dynamics, vol. 59, no. 4, pp. 588–612, 2019.

[5] Jin, Yin, and Chen, "Advanced estimation techniques for Vehicle System Dynamic State: A survey," Sensors, vol. 19, no. 19, p. 4289, 2019.