Is minimalistic MPC based on centroidal dynamics enough to race complex vehicles?

Author Names Omitted. Paper-ID [add your ID here]

***Abstract*—MPC online control problem enhanced in CasADi, a framework written by Andersson et al. [1].**

I. Introduction

II. Proposed approach

A. Vehicle model

B. MPC Model

In order to build an MPC internal model with its own dynamics, and which is as simple as possible, a point mass model is chosen. This model has 3 DOF and encloses in itself the fundamental parameters of the driven vehicle as mass, aerodynamic coefficients, grip and power limits for braking and traction phases. With this MPC internal model, the optimal control problem is formulated in order to find the optimal control sequence necessary to minimize the travel time of the next N meters of the track. In particular the control inputs of the point mass model are the total longitudinal force F_x, and the angular acceleration along the z-axes $r_p = dr/dt$, where r is the yaw rate. The point mass model dynamics is then formulated in spatial domain instead of the time one, and the *direct collocation* is used to transform the OCP into an NLP. The NLP is coded in a scripting environment using the Matlab interface to the open-source CasADi framework [1], which provides building blocks to efficiently formulate and solve large-scale optimization problems, and solved through IPOPT **cita ipopt**.

Once the problem is solved an optimal control sequence for the steering angle has to be extracted from the NLP solution. In order to do that we use the following formula

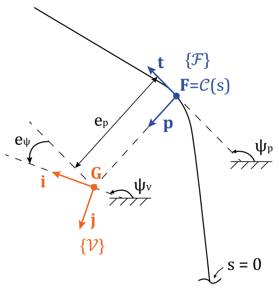
$$\delta = \alpha_1 + \beta_1 = F_{y_{11}}/C_\alpha + (v + ra_1)/u \qquad (1)$$

where the notation is based on **cita Guiggiani**, hence C_α is the cornering stiffness of the front tire, a_1 is the distance between the CoM and the front axle, and we assume that the tire slip angles and the vehicle slip angles are equal for the wheels of the same axle, hence $\alpha_{11} = \alpha_{12} = \alpha_1$ and $\beta_{11} = \beta_{12} = \beta_1$. Furthermore, the lateral force of the front wheels are extimated with a steady state assumption in a way that $F_{y_{11}} = F_{y_{12}} = F_y a_2/(2l)$, where F_y is the total lateral force acting on the point mass model, available from the NLP solution, a_2 is the distance between the CoM and the rear axle and l is the wheelbase.

For what concern the race track, it is assumed planar and modelled through the parametric 2D curve

$$\mathcal{C}(\alpha) = \{\boldsymbol{x}(\alpha) = [x(\alpha), y(\alpha)]^T \in \mathbb{R}^2 : \alpha \in [\alpha_0, \alpha_f]\} \qquad (2)$$

that identifies the road centerline, and the 1D curve $\mathcal{W}(\alpha)$ that specifies the track width. With reference to Figure 1a, the *curve parameter* α uniquely selects a point $\boldsymbol{F} = \boldsymbol{x}(\alpha)$ that defines the origin of the *Frenet-Serret frame* $\mathcal{F} = \{\boldsymbol{F}, (\boldsymbol{t}, \boldsymbol{p})\}$ whose unit vectors are, respectively, the tangent $\boldsymbol{t}$ and the normal $\boldsymbol{p}$ of the curve $\mathcal{C}$ in the point $\boldsymbol{F}$. The vehicle reference system $\mathcal{V} = \{\boldsymbol{G}, (\boldsymbol{i}, \boldsymbol{j})\}$ can be expressed in terms of the moving frame $\mathcal{F}$ with a *lateral displacement* e_p along the track normal direction $\boldsymbol{p}$ and the *heading error* e_ψ. In order to maintain $\mathcal{F}$ side-by-side with $\mathcal{V}$, the Frenet-Serret system has to proceed together with the vehicle: this leads to a relation between vehicle and Frenet-Serret velocities that ultimately imposes a bound between time and α increments.



(a)

Fig. 1. (**a**) vehicle pose respect to the Frenet-Serret reference system identified on the track curve.

C. Offset-free MPC

III. Preliminary results

The MPC controller is tested on Indianapolis oval track racing. In particular, three different simulation are performed: (i) lap-time simulation with MPC controller, (ii) lap-time simulation with MPC controller in which the aerodynamic drag is removed from the model equations, (iii) lap-time simulation with offset-free MPC, where the aerodynamic drag is estimated through the technique explained in **cita sez. offset-free**.

The solutions in terms of trajectories of the real vehicle and controls input provided to it, are shown below, for the entry phase of the first curve.

Figures.2 - 3 highlight how MPC with drag and MPC offset-free are able to drive the Adams vehicle in the same way, with a lap-time difference of 6.5ms. Instead if the aerodynamic drag is not modeled inside the MPC, the controller is not able to maintain the real vehicle on the track, in fact the simulation failed on the entry phase of the second curve.

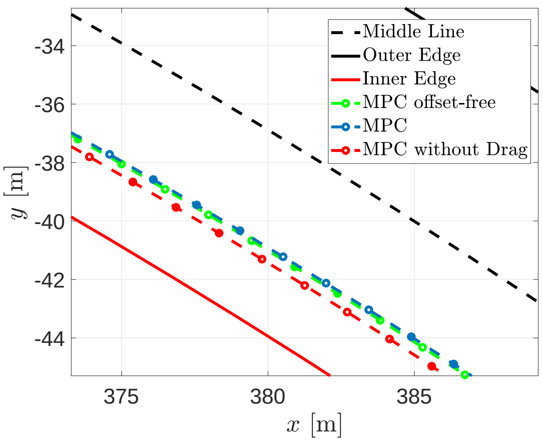


Fig. 2. Trajectories comparison between lap simulation with MPC (blue), MPC offset-free (green), MPC without drag (red)

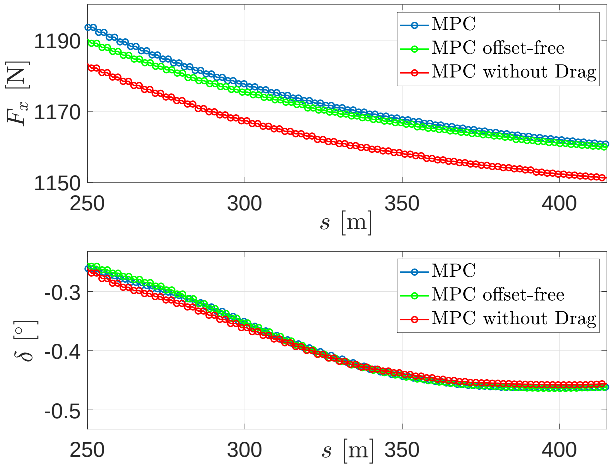


Fig. 3. Comparison between control inputs, longitudinal force (top), steering angle of the front wheel (bottom), obtained with MPC (blue), MPC offset-free (green) and MPC without drag (red)

IV. Conclusion

The minimalistic MPC based on a point mass model with its body-fixed reference system is enough to race complex vehicle. In particular the parameters of the MPC model as the mass, the aerodynamic coefficients or power limits, have to be equal to those of the driven car. In this way we can consider the point mass model as a condensed model of the complex vehicle.

Furthermore, the offset-free technique is an helpful and efficient method in order to modeled in a simple way some fundamental characteristics of the complex vehicle. As an example we consider not knowing the aerodynamic drag coefficient and we estimate drag force with the disturbance d.

As a future developments of this work, some modifies to the MPC internal model, as a new approach to extract the steer angle, has to be found in order to drive the complex car on its limits.

References

[1] Joel A.E. Andersson, Joris Gillis, Greg Horn, James B. Rawlings, and Moritz Diehl. CasADi: a software framework for nonlinear optimization and optimal control. *Math. Program. Comput.*, 11(1):1–36, mar 2019. ISSN 18672957. doi: 10.1007/s12532-018-0139-4. URL https://link.springer.com/article/10.1007/s12532-018-0139-4.