I Shapovalov¹,², S Gafurov¹ and A Klimchik¹

¹Center for technologies in Robotics and Mechatronics components, Innopolis University, 1 Universitetskaya Str., Innopolis, Russia
²Author to whom any correspondence should be addressed

Abstract. Control of self-driving vehicles is one of the up-to-date tasks. Vehicles are nonlinear dynamic systems with a large number of parameters. When it is supposed to move on a rough road with a lot of abrupt turns, significant unpredictable and unmeasurable disturbances emerge and the control quality degrades significantly. We implemented model predictive path integral (MPPI) control to deal with significant unmeasured disturbances. To approve efficiency of the proposed controller, we conducted simulation of complex car motion on a plane under the control of PID, and MPPI controllers. Performance of controllers is comparatively equal in stable modes, when reference position and velocity do not change. MPPI controllers outperform PID ones in unstable modes due to their ability to predict a vehicle behavior. Also, MPPI is the best choice for the rejection of significant external disturbances.

1. Introduction

Self-driving cars navigation is provided by a number of subsystems such as localization, routing and path-planning subsystems as well as a control subsystem and a perception [1]. Autonomous car control, route generation and path-planning are provided by a number of methods and algorithms. And usually, these tasks are considered separately. Moreover, oftentimes path-planning algorithms do not take into account a car dynamics model as there is a number of scenarios where it can be left out of the account. In such scenarios, a car kinematics model is only used for path-planning and grid/graph search techniques are used for global routing [2], [3]. However off-road driving scenarios as well as driving in an urban environment with a number of maneuvers require a highly-detailed dynamic model of the car in order to perform low-level path-planning and to ensure its feasibility in real-time.

Methods based on Model Predictive Control (MPC) and PID control are widely used for self-driving cars. MPC is a model-based approach which has shown good results for car navigation in real-time [4], [5]. MPC allows adjusting vehicle parameters according to driving conditions. It requires knowing of several operational modes and corresponding internal models of a vehicle to shift between models and adjust parameters of the models if it is necessary [6], [7]. Linearization of a non-linear model was performed in research [8] in order to control a vehicle that moves at a constant speed. The control is based on the nested structure of two PID regulators. The obtained results showed that the proposed approach is better than the standard MPC method [9].

However self-driving vehicles require a new type of control - integral control of its various subsystems in a case when vehicle dynamic properties are crucial to be considered. The Model Predictive Path Integral control method has been shown to have more potential for self-driving car implementation [10]. It allows considering the path-planning and control subsystems as a whole. MPPI controllers require a raster costmap for environment representation. The research [11] has investigated the efficiency of the MPPI algorithm for relatively simple driving conditions. Our
research focuses on the investigation of the MPPI method efficiency and its working performances compared to standard control methods.

2. Problem statement

We consider an electric car Kia Soul EV as a control object in this paper. The problem statement is to move autonomously on a rough surface track with several turns and short straight segments similar to country roads. At the same time, we suppose the road surface inclination is always zero. The motion speed varies in a range from 25 to 40 km/h. Such requirements to motion predetermine strong influence of various external disturbances that can be represented as a combination of step, sine and Gaussian-noise signals.

Planning the route of a vehicle is another complex control problem that deserves close attention and efforts that are outside the scope of the paper. We suppose that some planning subsystem forms the reference speed $V_{ref}$, lateral position $y_{ref}$ and car orientation $\psi_{ref}$ in a road coordinate system for the motion controller. The planner solves all problems related to providing trajectories for obstacles and potholes avoidance.

To describe motion of the self-driven car we use two coordinate frames as it is shown in figure 1. The first frame is vehicle-fixed and the second one is an earth-fixed coordinate frame. The origin of the vehicle-fixed frame is in the car centre of mass.

3. Model of vehicle

Most of research papers about self-driving vehicles discuss a single-track dynamics model [12] as it allows describing basic car dynamics without complex expressions with a large number of parameters. On the other hand, dual-track models allow considering a number of non-linearities occurring in extreme or close-to-extreme operational modes. The second type of models includes a number of parameters and their integration requires a great amount of calculation. That makes it inexpedient to implement such models in onboard computers to achieve control goals. But simulation of dual-track models on stationary computers provides an estimation of motion control and signal processing systems quality with the accuracy necessary for real-world problems.

Let us consider a vehicle with two axles for the description of its longitudinal and lateral motion with orientation variation. The schematic representation of the vehicle with used coordinate frames is given in figure 1.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{figure1.png}
\caption{Coordinate frames used for vehicle description.}
\end{figure}

Dynamics of a vehicle as a solid body can be represented in the form of the following equations [13]:

\[
\begin{align*}
\dot{x} &= \dot{y} r + \frac{F_{xfl} + F_{xfr} + F_{xrl} + F_{xrr} + F_{xext}}{m} \\
\dot{y} &= \dot{x} r + \frac{F_{yfl} + F_{yfr} + F_{ylr} + F_{yrr} + F_{yext}}{m} \\
\dot{r} &= \frac{a(F_{yfl} + F_{yfr}) b(F_{yrl} + F_{yrr})}{2(F_{xfl} - F_{xfr})} + \frac{w_f}{2(F_{xfl} - F_{xfr})} + \frac{w_r}{2(F_{xfl} - F_{xfr})} + \frac{M_d}{I_{zz}}
\end{align*}
\]

where $F_{xfl}$, $F_{xfr}$, $F_{xrl}$, $F_{xrr}$, $F_{xext}$, $F_{yfl}$, $F_{yfr}$, $F_{ylr}$, $F_{yrr}$, $F_{yext}$, $w_f$, $w_r$, $M_d$, and $I_{zz}$ are forces and moments acting on the vehicle, and $a$ and $b$ are coefficients depending on the vehicle's geometry.
where $m$ is the vehicle mass; $a$, $b$ are the distances to the front and rear axle correspondingly from the point of CM projection on the axle plane; $w_f$, $w_r$ are the front and rear track widths; $I_{zz}$ is the vehicle body moment of inertia about the vehicle-fixed $z$-axis.

Forces applied to the attachment points of wheels are defined as:

$$
F_{xft}=F_{xfr}=F_{xrt}=F_{xrr}=F_{xt} = F_{xrt},
$$

$$
F_{yft}=-C_yt\alpha t, F_{yfr}=C_y t \alpha t, F_{yrt}=-C_yt \alpha t, F_{yrr}=C_y t \alpha t
$$

where $F_{input}$ are the tractive forces emerging as a result of contact between the wheels and earth surface; $F_{znom}$ is the nominal normal force applied to the vehicle axles used to estimate friction parameters during weight and load shift; $C_y$, $\mu$ is the cornering stiffness of wheels, $\mu$ is the wheel friction coefficient.

Slip angles $\alpha$ are defined based on the ratio of velocities in the vehicle-fixed coordinate frame and the front wheels steering angle:

$$
\alpha_f = \tan \left( \frac{\dot{y} + \alpha r}{\dot{x} + \alpha r} \right), \alpha_r = \tan \left( \frac{\dot{y} + \alpha r}{\dot{x} + \alpha r} \right), \alpha_t = \tan \left( \frac{\dot{y} - \alpha r}{\dot{x} - \alpha r} \right)
$$

where $\delta_t$ is the steering angle of the front wheels.

The forces from equation (2) are included into equation (1) after their representation in the vehicle-fixed coordinate frame:

$$
F_{xt} = F_{xft} \cos (\delta_t) - F_{yft} \sin (\delta_t), F_{yt} = -F_{xfr} \sin (\delta_t) + F_{yfr} \cos (\delta_t), F_{xt} = F_{xrt}, F_{yt} = F_{yrt}
$$

4. Motion controller

Further, we give basic expressions needed to calculate MPPI control signals. To achieve better understanding of their inference, a reader can acquaint himself with [10].

Let’s consider nonlinear system in the form

$$
\dot{x}(t) = f(x(t), t) + G(x(t), t)u(t) + B(x(t), t)dw(t)
$$

The initial state vector of the system (5) is denoted as $x_0$ and the terminal state at time $T$ is $x_T$. It should be noted that the system is affine in control and Brownian disturbance. We assume, that $G$ and $B$ are partitioned in such a way that we can consider actuated and non-actuated dynamics of the system. Also, there should be no correlation between noise affecting actuated and non-actuated parts of the system (5) with a stationary noise in the actuated part:

$$
G = \begin{pmatrix} 0 \\ G_c \end{pmatrix}, B(x_c) = \begin{pmatrix} B_u(x_c) \\ 0 \end{pmatrix}
$$

The disturbance can be represented as

$$
dw(t) = e\sqrt{\Delta t}
$$

In discrete form it looks like

$$
\dot{x}_{ij} = f(x_{ij}, t_j) + G(x_{ij}, t_j)u_j \Delta t + B(x_{ij}, t_j)\epsilon \sqrt{\Delta t}
$$

The cost function for estimation of control quality over the system trajectory is

$$
\bar{S}(x, T) = \varphi(x_T) + \sum_{j=0}^{N-1} \varphi(x_j, u_j, \epsilon_j, t_j) \Delta t
$$

where $\varphi(x, T)$ is the terminal cost.

The second term in the right part of equation (7) is the sum of running costs consisting of control, state-dependent and stochastic components. We calculate the running costs according to the following expression

$$
\bar{g}(x_j, u_j, \epsilon_j, t) = g(x_j, t) + \frac{1}{2} u_j^T R_u u_j + \lambda \epsilon_j^T G \epsilon \frac{\epsilon}{\sqrt{\Delta t}} + \frac{1}{2} \lambda \epsilon_j^T G \epsilon \frac{\epsilon}{\sqrt{\Delta t}} B_u^T(B_u B_u^T)^{-1} B_u \epsilon
$$

where $R = R(x, t)$ is a positive definite matrix, $\lambda$ is a constant establishing the relation between noise and the cost function $S$.

We take the state-dependent part of the running cost as

$$
q(x, t) = k(x - x_{ref})^2
$$

The iterative form of the control signal $u_j$ is as follows
\[ u^*_j = u_j + \mathcal{H}^{-1} \mathcal{G} \left( E_{q'_u} \left[ \exp \left( \frac{1}{2} \mathcal{S} \left( \tau \right) \right) \frac{u_j}{E_{q'_u}} \exp \left( \frac{1}{2} \mathcal{S} \left( \tau \right) \right) \right] \right) \] (10)

The terms

\[ \mathcal{H} = G_c^T (B_c B_c^T)^{-1} G_c \quad \mathcal{G} = G_c^T (B_c B_c^T)^{-1} B_c \]

If we approximate the expectation \( E_{q'u} \) over a stochastic process \( q \) with a non-zero control input \( u \) and a changed variance \( v \) by a weighted average over sampled trajectories \( \tau_i \) and take the dependency \( \mathcal{H}^{-1} \mathcal{G} = \gamma \), the control (10) has the form

\[ u(x_{t_i}, t_i)^* \approx u(x_{t_i}, t_i) + \sum_{k=1}^{K} \exp \left( -1 \frac{1}{\lambda} \mathcal{S} \left( \tau_{i,k} \right) \right) \delta u_{i,k} \]

The algorithm of MPPI control describing a consequence of calculations for obtaining the controller output signal can be found in [8].

5. Simulation
Efficiency of presented approach was checked in MatLab environment. As the parameters of the model (1) we used the exact or approximate parameters of Kia Soul EV given in table 1.

| Parameter                  | Value  |
|---------------------------|--------|
| Mass \( m \), kg          | 1513   |
| Moment of inertia \( I_{zz} \), kg\( \cdot m^2 \) | 2066   |
| Distance to front axle \( a \), m | 1.2    |
| Distance to rear axle \( b \), m | 1.37   |
| Height of center of mass \( h \), m | 0.15   |
| Nominal normal force \( F_{znom} \), N | 7500   |
| Front wheel cornering stiffness \( C_{yf} \), N/\( \text{rad} \) | 15000  |
| Rear wheel cornering stiffness \( C_{yr} \), N/\( \text{rad} \) | 35000  |

The tire friction coefficient \( \mu \) has the value of 0.72. We also add several restrictions on the vehicle dynamics. The steering angle rate of change is 6 degrees per sec with a corresponding sign. The speed rate of change is 10 m/sec2 with a sign corresponding to acceleration or deceleration mode.

We compared performance of the PID and MPPI controllers in various scenarios to reveal their advantages also with restrictions. For the MPPI controller we used the prediction horizon of 1 sec with 50 sample trajectories.

We presented results of using the PID and MPPI controllers during motion without external disturbances and rare changes in reference signals in figure 2. Figure 2a demonstrates changing of the vehicle longitudinal speed, and figure 2b shows the trajectories obtained by two compared control systems. As it can be seen the performance of controllers is similar. MPPI is faster a little, but by the price of overshooting in the lateral coordinate.
Next, we tried to model the conditions in which the vehicle moves on the rough country road. For that we included external disturbances in the vehicle model in the form of white noise of significant amplitude comparing with the vehicle throttle. Also, we used a sine form of the reference signal for the lateral vehicle position. Obtained results are presented on figure 3, figure 4 and figure 5. Special attention should be given to the results in figure 5 where we presented unfiltered throttle and steering control of MPPI in presence of disturbance. It is obvious that without post processing of such signal control could damage powertrain of the vehicle.

Trajectories in figure 4 demonstrate efficiency of MPPI control in transition modes as it can follow reference appropriately with small delay.

![Figure 2](image2.png)

**Figure 2.** Comparison of PID and MPPI controllers in mode without disturbances.

![Figure 3](image3.png)

**Figure 3.** Longitudinal and lateral speed of vehicle in presence of disturbance.

![Figure 4](image4.png)

**Figure 4.** Trajectories of vehicle in presence of disturbance.
6. Conclusions and future work
Separate considering of self-driving car subsystems may lead to physically invalid routes as actual dynamics characteristics of the car are not taken into account. Developed in this research, the control method based on MPPI provides better behaviour of the vehicle. However considering of path-planning and control subsystems as a whole arises the problem of computational complexity and time delays that will be take into account in our further work.

7. Acknowledgments
This research was supported by the Russian Education Ministry under the Agreement No. 075-10-2018-017 on granting a subsidy from 11.26.2018 “Development of a commercial urban transport with the intellectual driver assistance system “City Pilot”. The customer is a federal target program. The unique identifier of the project RFMEFI60918X0005. Industrial partner – KAMAZ.

8. References
[1] Chaaban J 2014 A Methodology for designing robust central/self-organising multi-agent systems Int. J. of Computer Information Systems and Industrial Management Applications 6 571-81.
[2] LaValle S M and Branicky M S 2004 On the relationship between classical grid search and probabilistic roadmaps Algorithmic Foundations of Robotics V. Springer Tracts in Advanced Robotics vol 7, ed. Boissonnat JD., Burdick J., Goldberg K., Hutchinson S. (Berlin, Heidelberg: Springer).
[3] Latombe J C 1991 Robot Motion Planning (Boston, MA: Kluwer Academic Publishers).
[4] Khan M, Tahiyat M, Imtiaz S, Shoukat Choudhury M A A and Khan F 2017 Experimental evaluation of control performance of MPC as a regulatory controller ISA Transactions 70 512-20.
[5] Worthmann K, Mehrrez M W, Mann G K I, Gosine R G and Pannek J 2017 Interaction of open and closed loop control in MPC Automatica 82 243-50.
[6] Pacejka H 2004 Tire and Vehicle Dynamics (Oxford: Butterworth Heinemann).
[7] Choi M, Oh J J and Choi S B 2013 Linearized Recursive Least Squares Methods for Real-Time Identification of Tire–Road Friction Coefficient IEEE TRANS. ON VEHICULAR TECHNOLOGY 762 2906-18.
[8] Marino R, Scalzi S and Netto M 2011 Nested PID steering control for lane keeping in autonomous vehicles Control Engineering Practice 19 1459-7.
[9] MacAdam C 1981 Application of an optimal preview control for simulation of closed-loop automobile driving IEEE Transactions on Systems, Man, and Cybernetics 11 393-9.
[10] Williams G, Drews P, Goldfain B, Rehg J M and Theodorou E A 2016 Aggressive Driving with Model Predictive Path Integral Control // 2016 IEEE International Conference on Robotics and Automation (ICRA), pp. 1433 – 40.
[11] Buyval A, Gabdullin A and Klimchik A 2018 Model Predictive Path Integral Control for Car Driving with Dynamic Cost Map. Proc. 15th International Conference on Informatics in Control, Automation and Robotics (Porto) vol 1 pp 248-54.

[12] Ackermann J 2002 Robust Control (London: Springer).

[13] Gillespie T 1992 Fundamentals of Vehicle Dynamics (Warrendale, PA: SAE, Inc.).