Optimal Kinematic-based Trajectory Planning and Tracking Control of Autonomous Ground Vehicle Using the Variational Approach

Keyvan Majd[1], Mohammad Razeghi-Jahromi[2], and Abdollah Homaifar[1]

***Abstract*— In this paper, a novel kinematic-based optimal trajectory planning formulation for an autonomous vehicle is presented. In this new formulation, the quadratic errors of position, velocity, and acceleration are minimized subject to the rear wheel car-like vehicle nonlinear kinematic model. Minimizing the error of velocity and acceleration in addition to the error of position, allows us to obtain both optimal vehicle trajectory and control law. The Variational approach is used to minimize the cost function. Then, optimal trajectory and control inputs are numerically calculated by solving a set of two-point boundary value (TPBV) nonlinear differential equations. Finally, the proposed method is evaluated in two scenarios of lane changing and multi-curvature road which verify the success of the proposed method in generating an optimal trajectory and control inputs.**

I. INTRODUCTION

Motion planning is one of the key components of autonomous driving which deploys a variety of knowledges from different areas such as machine learning, artificial intelligence, control, and so on. Motion planning can be divided into the following four categories [1]:

- **Route Planning** is investigating a global route from origin to destination considering the road network [2]. A* [3] and Dijkstra [4] are among two most common algorithms for route planning.
- **Path Planning** is searching for a geometric path connecting the initial configuration to the terminal configuration satisfying the road boundaries and collision avoidance. Rapidly exploring random trees (RRT) [5], Lattice Planner [6], and Potential fields [7] are among the popular methods in this area.
- **Maneuver Planning** is deciding and choosing the safest maneuver with the lowest risk, considering interactions with the surrounding environment. Using decision making techniques [8] in collision avoidance [9] and driving assistance systems [10] are some examples of maneuver planning.
- **Trajectory Planning** is real-time transition planning of the vehicle which provides safe maneuver and kinematically feasible trajectory for the vehicles control system.

[1]Keyvan Majd and Abdollah Homaifar are with the department of Electrical and Computer Engineering and the Autonomous and Information Technology Institute, North Carolina A&T State University, Greensboro, NC 27403, USA kmajd@aggies.ncat.edu, homaifar@ncat.edu.

[2]Mohammad Razeghi-Jahromi is Research Scientist Digitalization for Power Conversion and Control Systems at ABB Corporate Research, Raleigh, NC, USA mohammad.razeghi-jahromi@us.abb.com.

The scope of this research is trajectory planning of autonomous vehicles. Many algorithms and methods are proposed in the literature addressing this issue.

In the conventional planning methods, to guarantee smoothness and safety of the trajectory, it is represented by a specific geometric function such as the Bezier curve [11], Spline [12], or polynomial [13] . In [14], the trajectory is generated by solving an optimization inside a driving corridor that minimizes the error of position, velocity, acceleration, jerk, and the yaw rate which is subject to the desired behavior of the vehicle given by the nonlinear inequality constraints. Nonetheless, these methods fails to generate feasible trajectories for the vehicle to follow as they do not consider the model of the vehicle and motion constraints.

In more recent planning methods, the vehicle system model is taken into account to guarantee the compatibility of the generated trajectory with the kinematics and nonholonomic constrains of the vehicle [15], [16]. Moreover, The authors in [17] developed a new trajectory tracking method in a kinematic-dynamic cascade structure where the control inputs are represented by a polynomial function. However, considering the linear kinematic model in [15], [16] and restricting the control input to polynomial representation in [17] reduce the solution space and cause suboptimality in these studies.

Even though these results are important, we take a broader look by (i) considering the nonlinear kinematic model of the vehicle, (ii) combining trajectory planning and tracking control together, and (iii) not restricting the trajectory and control inputs to the certain types of geometric function representations (e.g. Bezier curves, splines, and polynomials), which generally are neglected by the existing works.

In this study, a novel cost function is presented which gives a kinematically feasible optimal trajectory by minimizing the quadratic error of position, velocity, and acceleration subject to the nonlinear rear-wheel car-like vehicle model. The variational approach is used to solve the optimization and the corresponding optimal control law is obtained. At the end, the optimal trajectory and control inputs are found by solving a set of two-point boundary value (TPBV) nonlinear differential equations numerically. The proposed framework is capable of tracking any arbitrary reference with continuous acceleration profile, which is shown in the simulation part.

The remainder of this paper is organized as follows: section II formulates the model and problem definition. The proposed solution method is explained in section III. In section IV, the numerical results are described. Finally, the conclusions and future work are presented in section V.

II. PROBLEM FORMULATION

A. Model

In this paper, the kinematics of a rear wheel car-like vehicle model is used in trajectory generation and tracking control design which can be expressed as follows [18]

$$\begin{aligned}\dot{x}(t) &= v(t)\cos(\theta(t))\\ \dot{y}(t) &= v(t)\sin(\theta(t))\\ \dot{\theta}(t) &= \frac{1}{\ell}v(t)\tan(\phi(t))\\ \dot{v}(t) &= a(t)\end{aligned} \tag{1}$$

where t refers to time, (x, y) are longitudinal and lateral positions of the rear-wheel axis midpoint, θ is heading angle and $\phi \in (-\frac{\pi}{2}, \frac{\pi}{2})$, v, a, and ℓ are steering angle, velocity, acceleration and wheelbase length, respectively (see Fig. 1).

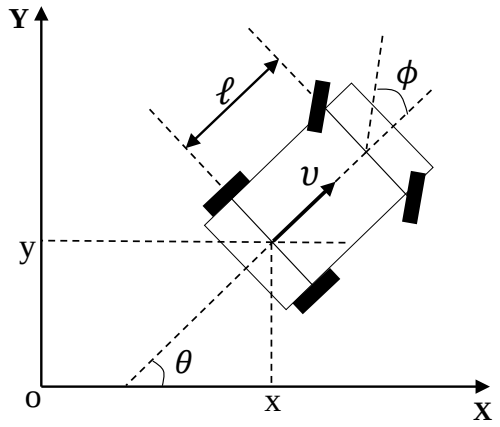

Fig. 1. Rear-wheel car-like vehicle model

B. Trajectory Planning

The state-space model (1) includes 4 states $\mathbf{x}(t) = [x(t)\ y(t)\ \theta(t)\ v(t)]^T$ and 2 control inputs $\mathbf{u}(t) = [\phi(t)\ a(t)]^T$. The trajectory planner computes the optimal trajectory $[x(t)\ y(t)]^T$ that minimizes

$$\begin{aligned}J(\mathbf{u}(t)) = \int_{t_0}^{t_f} \Big(& e_p^2(x(t), y(t), x_r(t), y_r(t))\\ & + e_v^2(\dot{x}(t), \dot{y}(t), \dot{x}_r(t), \dot{y}_r(t))\\ & + e_a^2(\ddot{x}(t), \ddot{y}(t), \ddot{x}_r(t), \ddot{y}_r(t))\Big) dt\end{aligned} \tag{2}$$

where

$$\begin{aligned}e_p^2(\cdot) &= \Big(x(t) - x_r(t)\Big)^2 + \Big(y(t) - y_r(t)\Big)^2\\ e_v^2(\cdot) &= \Big(\dot{x}(t) - \dot{x}_r(t)\Big)^2 + \Big(\dot{y}(t) - \dot{y}_r(t)\Big)^2\\ e_a^2(\cdot) &= \Big(\ddot{x}(t) - \ddot{x}_r(t)\Big)^2 + \Big(\ddot{y}(t) - \ddot{y}_r(t)\Big)^2\end{aligned} \tag{3}$$

where $e_p^2(\cdot)$, $e_v^2(\cdot)$, and $e_a^2(\cdot)$ are quadratic errors of position, velocity, and acceleration, respectively. Also, t_f is the fixed final time, $[x_r(t)\ y_r(t)]^T$ is the geometric reference trajectory which both are given by the path planner and is assumed to be at least twice differentiable.

Taking the derivative of $\dot{x}(t)$ and $\dot{y}(t)$ in (1), we have

$$\begin{aligned}\ddot{x}(t) &= a(t)\cos(\theta(t)) - \frac{1}{\ell}v^2(t)\tan(\phi(t))\sin(\theta(t))\\ \ddot{y}(t) &= a(t)\sin(\theta(t)) + \frac{1}{\ell}v^2(t)\tan(\phi(t))\cos(\theta(t))\end{aligned} \tag{4}$$

substituting $\dot{x}(t)$, $\dot{y}(t)$, $\ddot{x}(t)$, and $\ddot{y}(t)$ in (3) gives

$$\begin{aligned}e_p^2(\cdot) &= \Big(x(t) - x_r(t)\Big)^2 + \Big(y(t) - y_r(t)\Big)^2\\ e_v^2(\cdot) &= \Big(v(t)\cos(\theta(t)) - \dot{x}_r(t)\Big)^2 + \Big(v(t)\sin(\theta(t)) - \dot{y}_r(t)\Big)^2\\ e_a^2(\cdot) &= \\ &\Big(a(t)\cos(\theta(t)) - \frac{1}{\ell}v^2(t)\tan(\phi(t))\sin(\theta(t)) - \ddot{x}_r(t)\Big)^2\\ &+ \Big(a(t)\sin(\theta(t)) + \frac{1}{\ell}v^2(t)\tan(\phi(t))\cos(\theta(t)) - \ddot{y}_r(t)\Big)^2\end{aligned} \tag{5}$$

The terms $e_v^2(\cdot)$ and $e_a^2(\cdot)$ are added in the cost function to guarantee the continuity of velocity and acceleration by tracking the first and the second derivatives of reference trajectory. Moreover, it allows us to extract the optimal control $\mathbf{u}^*(t)$ since the control inputs appear explicitly in the cost function.

As a result, the goal is for $[x(t)\ y(t)]^T$ to follow $[x_r(t)\ y_r(t)]^T$ and the reference trajectory is also assumed to be twice differentiable; thus, $[\dot{x}(t)\ \dot{y}(t)]^T$ and $[\ddot{x}(t)\ \ddot{y}(t)]^T$ should follow $[\dot{x}_r(t)\ \dot{y}_r(t)]^T$ and $[\ddot{x}_r(t)\ \ddot{y}_r(t)]^T$, respectively.

C. Boundary Conditions

Since the path planner initiates its pathfinding from the initial coordination of the vehicle and our goal is to track the desired reference path, the initial and final position of the vehicle should be on the reference trajectory, i.e. $(x(t_0), y(t_0)) = (x_r(t_0), y_r(t_0))$ and $(x(t_f), y(t_f)) = (x_r(t_f), y_r(t_f))$. Considering the fact that the error $e_v^2(\cdot)$ is expected to be zero at final time t_f,

$$\begin{aligned}\dot{x}(t_f) &= v(t_f)\cos(\theta(t_f))\\ \dot{y}(t_f) &= v(t_f)\sin(\theta(t_f))\end{aligned} \tag{6}$$

$\theta(t_f)$ and $v(t_f)$ can also be found as

$$\begin{aligned}\theta(t_f) &= \arctan\left(\frac{\dot{y}(t_f)}{\dot{x}(t_f)}\right)\\ v(t_f) &= \begin{cases}\frac{\dot{x}(t_f)}{\cos(\theta(t_f))}, & \theta(t_f) \neq \pm\frac{\pi}{2}\\ \frac{\dot{y}(t_f)}{\sin(\theta(t_f))}, & \theta(t_f) \neq 0\end{cases}\end{aligned} \tag{7}$$

Finally, θ_0 and v_0 follow the initial conditions of the vehicle.

III. SOLUTION METHOD

Lemma 1: An optimal control $\mathbf{u}^*(t) = [\phi^*(t)\ a^*(t)]^T$ for $t \in [t_0, t_f]$ which causes the system (1) to follow an optimal state $\mathbf{x}^*(t) = [x^*(t)\ y^*(t)\ \theta^*(t)\ v^*(t)]^T$ that minimizes the

cost function (2) is given by solving the following two-point boundary value (TPBV) state nonlinear differential equations

$$\begin{aligned}\dot{x}^*(t) &= v^*(t)\cos(\theta^*(t))\\ \dot{y}^*(t) &= v^*(t)\sin(\theta^*(t))\\ \dot{\theta}^*(t) &= \frac{1}{\ell}v^*(t)\tan(\phi^*(t))\\ \dot{v}^*(t) &= a^*(t)\end{aligned} \tag{8}$$

and co-state nonlinear differential equations

$$\begin{aligned}\dot{p}_1^*(t) &= -2(x^*(t)-x_r(t))\\ \dot{p}_2^*(t) &= -2(y^*(t)-y_r(t))\\ \dot{p}_3^*(t) &= \Big(2v^*(t)\dot{y}_r(t) - \frac{2}{\ell}\ddot{x}_r(t){v^*}^2(t)\tan(\phi^*(t))\\ &\quad + 2a^*(t)\ddot{y}_r(t) - p_2^*(t)v^*(t)\Big)\cos(\theta^*(t))\\ &\quad - \Big(2v^*(t)\dot{x}_r(t) + \frac{2}{\ell}\ddot{y}_r(t){v^*}^2(t)\tan(\phi^*(t))\\ &\quad + 2a^*(t)\ddot{x}_r(t) - p_1^*(t)v^*(t)\Big)\sin(\theta^*(t))\\ \dot{p}_4^*(t) &= \\ &\Big(2\dot{x}_r(t) + \frac{4}{\ell}v^*(t)\ddot{y}_r(t)\tan(\phi^*(t)) - p_1^*(t)\Big)\cos(\theta^*(t))\\ &+ \Big(2\dot{y}_r(t) - \frac{4}{\ell}v^*(t)\ddot{x}_r(t)\tan(\phi^*(t)) - p_2^*(t)\Big)\sin(\theta^*(t))\\ &- 2v^*(t) - \frac{4}{\ell^2}{v^*}^3(t)\tan^2(\phi^*(t)) - \frac{1}{\ell}p_3^*(t)\tan(\phi^*(t))\end{aligned} \tag{9}$$

with boundary conditions as shown in Section II-C. The optimal control laws are also given by

$$\begin{aligned}\phi^*(t) &= \arctan\Bigg(\frac{\ell}{{v^*}^3(t)}\Big(v^*(t)\ddot{y}_r(t)\cos(\theta^*(t))\\ &\quad - v^*(t)\ddot{x}_r(t)\sin(\theta^*(t)) - \frac{1}{2}p_3^*(t)\Big)\Bigg)\\ a^*(t) &= \ddot{x}_r(t)\cos(\theta^*(t)) + \ddot{y}_r(t)\sin(\theta^*(t)) - \frac{1}{2}p_4^*(t)\end{aligned} \tag{10}$$

Proof: First, we form the Hamiltonian as follows [19]

$$\begin{aligned}\mathcal{H}(\mathbf{x}(t),\mathbf{u}(t),\mathbf{p}(t)) &= e_p^2(\cdot) + e_v^2(\cdot) + e_a^2(\cdot)\\ &\quad + p_1(t)v(t)\cos(\theta(t))\\ &\quad + p_2(t)v(t)\sin(\theta(t))\\ &\quad + p_3(t)\Big(\frac{1}{\ell}v(t)\tan(\phi(t))\Big)\\ &\quad + p_4(t)a(t)\end{aligned} \tag{11}$$

where the vector $\mathbf{p}(t) = [p_1(t)\ p_2(t)\ p_3(t)\ p_4(t)]^T$ includes the Lagrange multipliers. The necessary conditions of optimality are [19]:

$$\dot{\mathbf{x}}^*(t) = \frac{\partial\mathcal{H}}{\partial\mathbf{p}}(\mathbf{x}^*(t),\mathbf{u}^*(t),\mathbf{p}^*(t)) \tag{12}$$

$$-\dot{\mathbf{p}}^*(t) = \frac{\partial\mathcal{H}}{\partial\mathbf{x}}(\mathbf{x}^*(t),\mathbf{u}^*(t),\mathbf{p}^*(t)) \tag{13}$$

$$0 = \frac{\partial\mathcal{H}}{\partial\mathbf{u}}(\mathbf{x}^*(t),\mathbf{u}^*(t),\mathbf{p}^*(t)) \tag{14}$$

Equations (8) and (9) are derived from substitution of $\mathcal{H}(\cdot)$ in the equations (12) and (13), respectively. Also, (14) gives

$$\begin{aligned}\frac{\partial\mathcal{H}(\cdot)}{\partial\phi} &= \frac{2}{\ell}v^*(t)\left(1+\tan^2(\phi^*(t))\right)\Big(\frac{1}{\ell}{v^*}^3(t)\tan(\phi^*(t))\\ &\quad + v^*(t)\ddot{x}_r(t)\sin(\theta^*(t)) - v^*(t)\ddot{y}_r(t)\cos(\theta^*(t))\\ &\quad + \frac{1}{2}p_3^*(t)\Big) = 0\\ \frac{\partial\mathcal{H}(\cdot)}{\partial a} &= a^*(t) - \ddot{x}_r(t)\cos(\theta^*(t)) - \ddot{y}_r(t)\sin(\theta^*(t))\\ &\quad + \frac{1}{2}p_4^*(t) = 0\end{aligned} \tag{15}$$

from which the optimal control laws in (10) are obtained. This completes the proof. ∎

Note that the $\arctan(\cdot)$ is a continuous function of its argument between $(-\frac{\pi}{2},\frac{\pi}{2})$; thus, in equation (10) the argument of $\arctan(\cdot)$ is continuous which implies continuous steering angle and consequently continuous curvature $\big(\kappa(t)\propto\phi(t)\big)$.

IV. NUMERICAL RESULTS

A. Numerical Approach for Solving the Two-point Boundary Value Problem

Solving the differential equations with fixed initial and final boundaries is named the two-point boundary value (TPBV) problem. In a nonlinear case, analytical solutions may be found for special examples of TPBV problems, but generally, these type of problems are solved numerically in the literature. The collocation method is one of the most popular techniques among numerous existing numerical solution methods for this type of problem. One of the major advantages of collocation methods is their insensitivity to the boundary values variations which is why we used it to solve our problem.

To solve the TPBV differential equations (8) and (9), we used BVP4C [20] package of MATLAB. This package solves some standard forms of TPBV problems by using the forth-order Lobatto IIIa formula. Since the problem is tracking, the reference trajectories are selected as initial guesses for the state variables. Moreover, the co-state variables are anticipated to become zero which is the assumed initial guess for these variables.

B. Simulation

The performance of the proposed method is evaluated in two different scenarios: lane changing and a multi-curvature road. The results of *Lemma 1* are used to generate the optimal vehicle trajectory and control inputs. To capture the effect of the vehicle kinematic model in the trajectory planning, $v(t_0)$ and $\theta(t_0)$ are chosen such that the initial quadratic errors of velocity $e_v^2(\cdot)$ and acceleration $e_a^2(\cdot)$ not to be zero. The simulations are developed in MATLAB.

1) Lane-change scenario: It is assumed that the reference trajectory and the final time are given by the path planner, where the initial time is $t_0 = 10\ sec$ and the final fixed time is $t_f = 70\ sec$. The reference trajectory is given as follows $t \in [t_0, t_f]$

$$y_r(x_r) = 5 + 2.5\tanh\left(\frac{x_r - 40}{10}\right) \tag{16}$$

Also, the wheelbase length of the vehicle is $\ell = 1m$. Table I shows the boundary conditions for this scenario.

TABLE I

BOUNDARY CONDITIONS FOR THE LANE-CHANGE SCENARIO

$x(t_0) = 10\ (m)$	$x(t_f) = 70\ (m)$
$y(t_0) = 2.5\ (m)$	$y(t_f) = 7.5\ (m)$
$\theta(t_0) = 0.15\ (rad)$	$\theta(t_f) = 0\ (rad)$
$v(t_0) = 5\ (\frac{m}{s})$	$v(t_f) = 1\ (\frac{m}{s})$

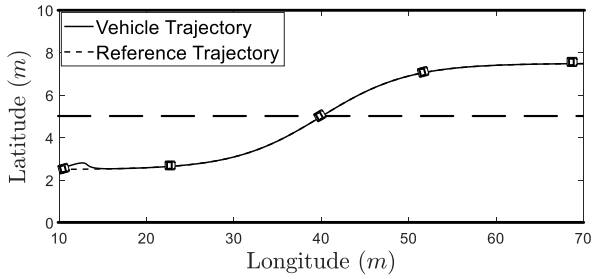

Fig. 2. Optimal vehicle trajectory (lane-change scenario)

The generated trajectories are demonstrated in Fig. 2. It can be seen that the tracking error converges to zero after 5 meters, and the vehicle trajectory meets the boundary conditions as mentioned in Table I. The transient shows that the vehicle cannot instantly track the reference trajectory due to the error between the initial configuration of the vehicle and the reference trajectory.

In Fig. 3, heading angle, velocity, and their references are shown which meet the boundary conditions as mentioned in Table I. The reference heading angle and velocity are derived from the flatness property of system (1) as follows

$$\theta_r(t) = \arctan\left(\frac{\dot{y}_r(t)}{\dot{x}_r(t)}\right)$$
$$v_r(t) = \begin{cases} \frac{\dot{x}_r(t)}{\cos(\theta_r(t))}, & \theta_r(t) \neq \pm\frac{\pi}{2} \\ \frac{\dot{y}_r(t)}{\sin(\theta_r(t))}, & \theta_r(t) \neq 0 \end{cases} \tag{17}$$

Moreover, the figure shows that the tracking occurs after 5 seconds.

Fig. 4 illustrates the optimal control inputs, steering angle and acceleration, given by (10) which makes the car with the kinematic model (1) follow the generated trajectory (16). As Fig. 4 (a) shows, the steering angle does not have any discontinuity and varies smoothly.

2) Multi-curvature scenario: Performance of the controller in a multi-curvature road is evaluated in this scenario. The initial time is $t_0 = 0\ sec$ and the final fixed time is $t_f = 20\ sec$. The reference trajectory for $t \in [t_0, t_f]$ is defined as

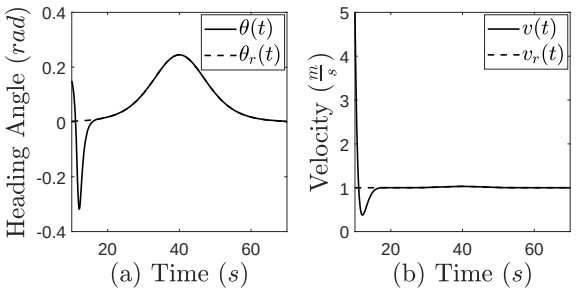

Fig. 3. Lane-change scenario: (a) heading angle: $\theta^*(t)$ and $\theta_r(t)$, (b) velocity: $v^*(t)$ and $v_r(t)$

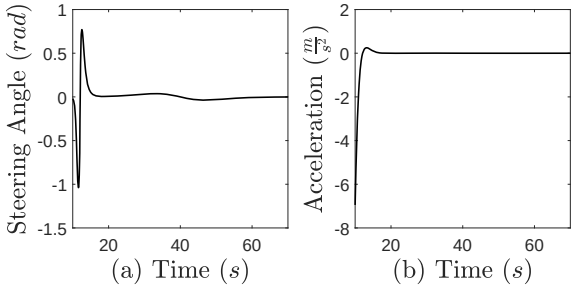

Fig. 4. Optimal control inputs (lane-change scenario): (a) steering angle: $\phi^*(t)$, (b) acceleration: $a^*(t)$

$$y_r(x_r) = 3.5\sin(\frac{2\pi}{15}x_r) - 4\cos(\frac{2\pi}{20}x_r) + 6 \tag{18}$$

The wheelbase length of the vehicle is $\ell = 1m$ and the boundary conditions are shown in Table II, following the results of Section II-C. Fig. 5 shows the trajectory of the vehicle. It can be seen that the tracking error becomes zero after almost 5 meters. Fig. 6 demonstrates heading angle and velocity in which the references are derived from (17). Fig. 7 illustrates the simulated control inputs: steering angle and acceleration.

TABLE II

BOUNDARY CONDITIONS FOR THE MULTI-CURVATURE SCENARIO

$x(t_0) = 0\ (m)$	$x(t_f) = 20\ (m)$
$y(t_0) = 2\ (m)$	$y(t_f) = 5\ (m)$
$\theta(t_0) = 1.3\ (rad)$	$\theta(t_f) = -0.63\ (rad)$
$v(t_0) = 5\ (\frac{m}{s})$	$v(t_f) = 1.24\ (\frac{m}{s})$

Same as the previous scenario, the transient occurs due to the consideration of the kinematic model in the minimization problem and a smooth steering angle is generated which guarantees the continuity of curvature.

V. CONCLUSIONS AND FUTURE WORK

In this work, a novel optimal trajectory planning and tracking control formulation is presented for the vehicle kinematic model.

Minimizing the quadratic errors of velocity and acceleration in addition to the position quadratic error in the cost

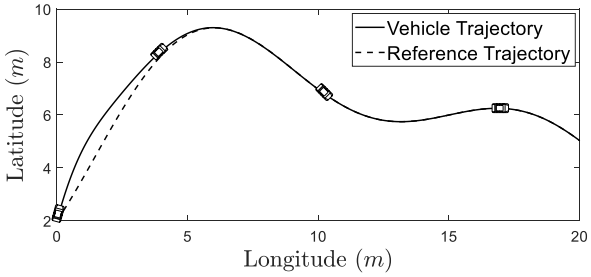

Fig. 5. Optimal vehicle trajectory (multi-curvature scenario)

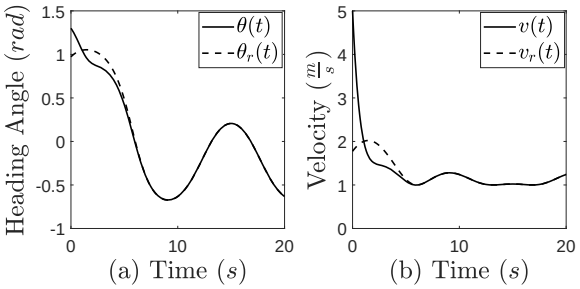

Fig. 6. Multi-curvature scenario: (a) heading angle: $\theta^*(t)$ and $\theta_r(t)$, (b) velocity: $v^*(t)$ and $v_r(t)$

function enables us to extract the optimal control $\mathbf{u}^*(t)$. Additionally, it guarantees the continuity of steering and curvature without considering any parametrized polynomial function representation for the steering angle. At the end, optimal vehicle trajectory and control inputs are numerically calculated by solving a set of TPBV nonlinear differential equations. The proposed algorithm provides the optimal trajectory for any arbitrary desired reference with a continuous acceleration profile.

In our future work, mechanical and physical constraints of the vehicle and boundaries of the road will be considered to achieve a safer and more comfortable navigation.

ACKNOWLEDGMENT

This work is partially supported by the US Department of Transportation (USDOT), Research and Innovative Technology Administration (RITA) under University Transportation Center (UTC) Program DTRT13-G-UTC47, and Air Force Research Laboratory and OSD under agreement number FA8750-15-2-0116.

REFERENCES

[1] C. Katrakazas, M. Quddus, W.-H. Chen, and L. Deka, "Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions," *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 416–442, 2015.

[2] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, 2016.

[3] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[4] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

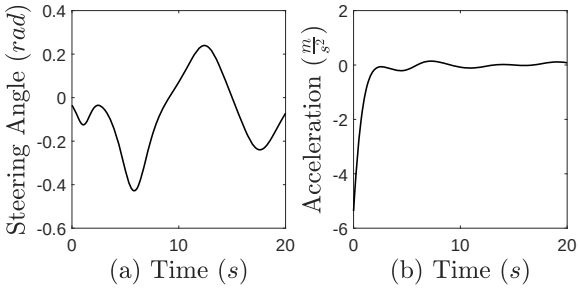

Fig. 7. Optimal control inputs (multi-curvature scenario): (a) steering angle: $\phi^*(t)$, (b) acceleration: $a^*(t)$

[5] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," 1998.

[6] M. Pivtoraiko and A. Kelly, "Efficient constrained path planning via search in state lattices," in *International Symposium on Artificial Intelligence, Robotics, and Automation in Space*, 2005, pp. 1–7.

[7] Y. Rasekhipour, A. Khajepour, S.-K. Chen, and B. Litkouhi, "A potential field-based model predictive path-planning controller for autonomous road vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1255–1267, 2017.

[8] A. Furda and L. Vlacic, "Enabling safe autonomous driving in real-world city traffic using multiple criteria decision making," *IEEE Intelligent Transportation Systems Magazine*, vol. 3, no. 1, pp. 4–17, 2011.

[9] Z. Wang, S. Ramyar, S. M. Salaken, A. Homaifar, S. Nahavandi, and A. Karimoddini, "A collision avoidance system with fuzzy danger level detection," in *Intelligent Vehicles Symposium (IV), 2017 IEEE*. IEEE, 2017, pp. 283–288.

[10] S. Ramyar, A. Homaifar, S. M. Salaken, S. Nahavandi, and A. Kurt, "A personalized highway driving assistance system," in *Intelligent Vehicles Symposium (IV), 2017 IEEE*. IEEE, 2017, pp. 1596–1601.

[11] D. González, J. Pérez, R. Lattarulo, V. Milanés, and F. Nashashibi, "Continuous curvature planning with obstacle avoidance capabilities in urban scenarios," in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*. IEEE, 2014, pp. 1430–1435.

[12] T. Gu and J. M. Dolan, "Toward human-like motion planning in urban environments," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*. IEEE, 2014, pp. 350–355.

[13] T. Shim, G. Adireddy, and H. Yuan, "Autonomous vehicle collision avoidance system using path planning and model-predictive-control-based active front steering and wheel torque control," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of automobile engineering*, vol. 226, no. 6, pp. 767–778, 2012.

[14] J. Ziegler, P. Bender, M. Schreiber, H. Lategahn, T. Strauss, C. Stiller, T. Dang, U. Franke, N. Appenrodt, C. G. Keller *et al.*, "Making bertha drive?an autonomous journey on a historic route," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 2, pp. 8–20, 2014.

[15] D. Madås, M. Nosratinia, M. Keshavarz, P. Sundström, R. Philippsen, A. Eidehall, and K.-M. Dahlén, "On path planning methods for automotive collision avoidance," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, pp. 931–937.

[16] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, "Kinematic and dynamic vehicle models for autonomous driving control design," in *Intelligent Vehicles Symposium (IV), 2015 IEEE*. IEEE, 2015, pp. 1094–1099.

[17] X. Li, Z. Sun, D. Cao, D. Liu, and H. He, "Development of a new integrated local trajectory planning and tracking control framework for autonomous ground vehicles," *Mechanical Systems and Signal Processing*, vol. 87, pp. 118–137, 2017.

[18] A. De Luca, G. Oriolo, and C. Samson, "Feedback control of a nonholonomic car-like robot," *Robot motion planning and control*, pp. 171–253, 1998.

[19] D. E. Kirk, *Optimal control theory: an introduction*. Courier Corporation, 2012.

[20] L. F. Shampine, J. Kierzenka, and M. W. Reichelt, "Solving boundary value problems for ordinary differential equations in matlab with bvp4c," *Tutorial notes*, vol. 2000, pp. 1–27, 2000.