Pure Pursuit Guidance for Car-Like Ground Vehicle Trajectory Tracking

Conference Paper · October 2017

DOI: 10.1115/DSCC2017-5376

2 authors:

Yuanyan Chen
Ohio University
2 PUBLICATIONS **0** CITATIONS
SEE PROFILE

Jim Zhu
Ohio University
135 PUBLICATIONS **1,487** CITATIONS
SEE PROFILE

Proceedings of DSCC 2017
the ASME 2017 Dynamic Systems and Control Conference
October 11-13, 2017, Tyson Coner, Virginia, USA

DSCC2017-5376

PURE PURSUIT GUIDANCE FOR CAR-LIKE GROUND VEHICLE TRAJECTORY TRACKING

Yuanyan Chen
Electric Engineering and Computer Science
Ohio University
Athens, Ohio 45701
Email: yc187911@ohio.edu

J.Jim Zhu*
Electric Engineering and Computer Science
Ohio University
Athens, Ohio 45701
Email: zhuj@ohio.edu

ABSTRACT

Trajectory tracking guidance and control for nonholonomic (car-like) Autonomous Ground Vehicles (AGV), such as self-driving cars and car-like wheeled mobile robots, is a more challenging control problem than path following control, because the latter does not impose a speed requirement on the vehicle motion. The tracking error dynamics along the nominal path are nonlinear and time-varying in nature, which need to be exponentially stabilized. This paper presents a Line-of-Sight (LOS) Pure-Pursuit Guidance (PPG) trajectory design algorithm that generates a three Degrees of Freedom (DOF) spatial trajectory for an AGV equipped with a 3DOF trajectory tracking controller. The LOS PPG can be used for cooperative, passive (neutral) and adversarial tracking tasks, such as, respectively, formation driving, autonomous lane keeping with speed requirement, and chasing an evading vehicle. The algorithm is verified with computer simulations on a 1/6 scale electric car model, and will be further validated on that model car in the near future.

INTRODUCTION

According to a statistic of 2014, there were 6 million police-reported highway collisions, which resulted in more than 2 million injuries and 32,675 fatalities [1]. Among all of these accidents, 94% were attributed to driver errors [2]. Autonomous Ground Vehicle (AGV) has the potential to reduce the number of fatal accidents and injuries due to driver errors. In the last three decades, there were increased research both in academics and industry towards developing AGV.

The SAE standard [3] defines 6 levels for vehicle automation. Level 0 represents no automation, where all driving tasks are the responsibility of the driver. Level 1 provides individual driver assistance, which includes adaptive cruise control, and anti-lock braking. Level 2 provides partial automation, with one or more driver assistance system of both acceleration and steering. Level 3 is conditional automation with low confidence. Level 4 is high automation with high confidence and level 5 is full automation. Our current research on vehicle trajectory tracking is to facilitate high levels of autonomy, level 3 and above.

Current conditional automation for ground vehicles mostly focuses on the path-following control objective, where the vehicle is required to converge to and follow a path that is specified without a temporal law. Current well-known methods for ground vehicle path-following include a particular type of pure pursuit guidance [4–6] and model predictive control [12–14].

On the other hand, trajectory tracking regulates the vehicle along a nominal path with time constraint, which requires the vehicle to not only follow the path, but also satisfy a specific speed profile. It is noted in [2] and [4], trajectory tracking is a nonlinear time varying control problem, which is a more challenging control task than path following. Lyapunov based design [15] and Output feedback linearization [16] have been applied in trajectory tracking control for nonholonomic (car-like) robots, but the results reported therein were only verified by simulation or validated on small scale mobile robots for simple trajectories at very low speed. Recently, a 3 Degrees of Freedom (DOF) nonlin-

*Address all correspondence to this author.

ear trajectory tracking controller using Trajectory Linearization Control (TLC) for car-like AGVs has been developed in [17], which takes into consideration the tire nonlinearity and vehicle nonlinear dynamics.

It is well-known that for nonlinear systems, the stability of motion is generally local in nature and the Domain of Attraction (DOA) is finite. So when the vehicle tracking error is out of the DOA, like when the initial vehicle state is far from the mission trajectory, then a guidance trajectory generator is necessary to design a feasible nominal trajectory to lead the vehicle to the mission trajectory. Figure 1 shows the complete ground vehicle Guidance, Navigation and Control (GN&C) system. In this paper, we will focus on trajectory generator design for a 3DOF postion trajectory tracking controller for car-like AGVs, such as the one in [17].

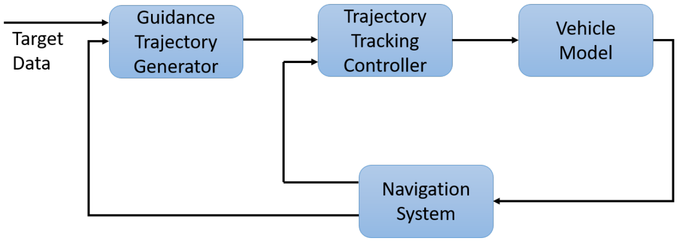

Figure 1. Diagram for Guidance Logic

Pure pursuit is well known for its effectiveness and simplicity as a guidance algorithm, which has been widely used in ground vehicle path-following from 1985, see [5–8]. The pure pursuit guidance (PPG) for ground vehicle path-following calculates a circumference arc that joins the current position of the vehicle and a target point in front of the vehicle on the nominal path. The vehicle then follows the arc as it moves forward towards the nominal path. Thus we will call this type of PPG as conventional ARC PPG, which does not consider the speed of the vehicle and is therefore not suitable for trajectory tracking. In [9], the ARC PPG takes velocity into consideration, which makes ARC PPG adaptable to trajectory tracking problem.

Another type of PPG has been used in aircraft guidance design, see [10], which aligns the velocity vector of the pursuing vehicle with the line-of-sight (LOS) joining the vehicle with a real or virtual target; the normal acceleration command is created and further converted into the attitude command. To distinguish this type of PPG from the ARC PPG, it will be referred to as LOS PPG. In [18] the LOS PPG was adapted as trajectory generator for aircraft in the inertial frame instead of generating normal acceleration commands to take advantage of the trajectory tracking controller.

Unlike aircraft, which enjoy full 6DOF motions, the car-like ground vehicles are subject to nonholonomic motion constraint,where the lateral force cannot generate lateral acceleration. So conventional LOS PPG which generate normal acceleration command cannot be used directly to ground vehicle guidance. We need to modify LOS PPG to generate a feasible trajectory for the tracking controller.

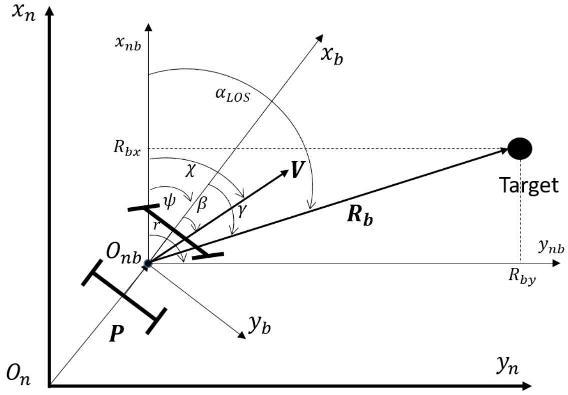

Figure 2. Position View of the UGV and the Target

In this paper, the LOS PPG will be adapted for the car-like ground vehicle trajectory tracking guidance. The contribution of this paper includes: 1) Applying LOS PPG to trajectory tracking instead of path-following for nonlinear, nonholonomic car-like vehicle; 2) Decouple the line-of-sight into heading and speed guidance control in the body frame to overcome the lateral nonholonomic constraint; 3) Generate a guidance trajectory in the inertial frame for the 3DOF inertial position trajectory controller instead of generating acceleration command.

The rest of the paper is structured as: in the section Trajectory Tracking Controller, a brief introduction of our trajectory tracking controller algorithm is presented. The Pure Pursuit Guidance Design section describes the complete PPG guidance algorithm. The Simulation and Results section presents the guidance algorithm on the 3DOF trajectory tracking controller as presented in [17]. The last section concludes the paper.

Trajectory Tracking Controller

The work presented in this paper is the continuation of Trajectory Linearization Control (TLC) applied on nonholonomic car-like AGVs. In order to facilitate the exposition of our main results on LOS PPG design, implementation and simulation testing, in this section we briefly review the 3DOF nonholonomic AGV trajectory tracking controller presented in [17]. In particular, we will explain how the b-frame lateral position error is regulated by rotational control in that controller, which is pertinent to how our LOS PPG handles the nonholonomic constraint for car-like AGVs. In general, our LOS PPG is applicable to any 3DOF tracking controller having similar capabilities of that

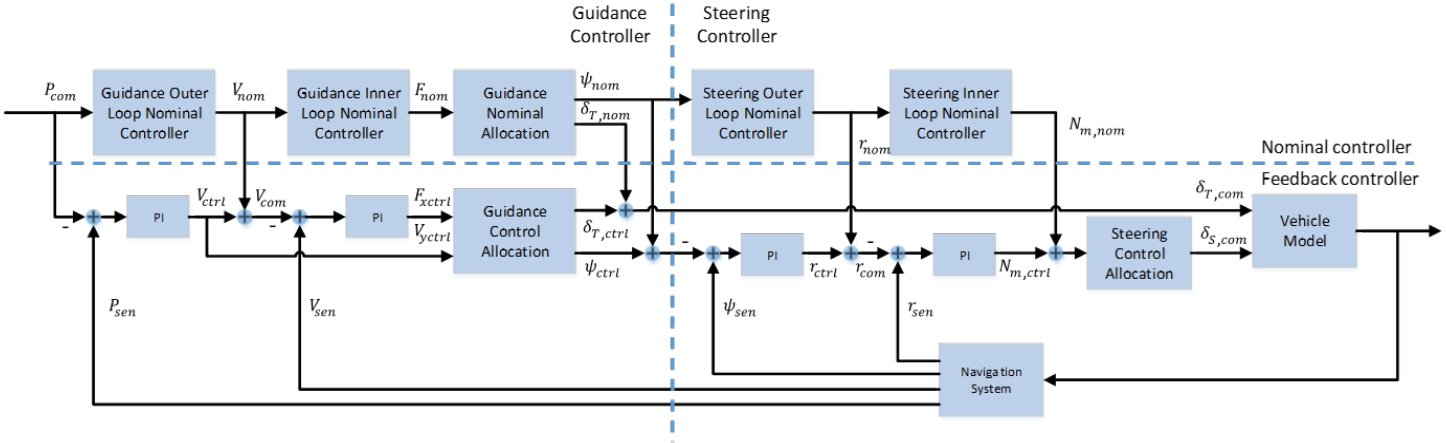

Figure 3. 3DOF Trajectory Linearization Controller Block Diagram

in [17].

Three reference frames are employed, a North-East-Down (NED) navigation frame (n-frame), an NED body-carried frame (nb-frame) and a body-fixed frame (b-frame). The n-frame's origin O_b is fixed at a point of interest on the surface of the Earth, which is treated as flat and inertial, with x_n pointing to the north, y_n pointing to the east, and z_n pointing down. The nb-frame is parallel to the n-frame but has the origin O_{nb} at center of the gravity (CG) of the vehicle. For the b-frame, the origin O_b is fixed at the CG of the vehicle, x_b along the longitude of the vehicle pointing forward, y_b pointing to the driver's right side of the vehicle and z_b pointing down. The relationship of the reference frames are shown in Figure 2, where β is the vehicle side-slip angle; χ is vehicle course angle; α_{LOS} is LOS angle in the n-frame, and γ is LOS angle in the b-frame; $P = \begin{bmatrix} x_n & y_n \end{bmatrix}^T$ is the vehicle position in the n-frame and R_b is the down range vector in the b-frame; $V = \begin{bmatrix} u & v \end{bmatrix}^T$ is the b-frame velocity; ψ is yaw angle; $r = \dot{\psi}$ is the the yaw angular rate.

TLC is a nonlinear time-varying controller, which combines nonlinear dynamic inversion with linear time-varying feedback stabilization. It can be viewed as the gain-scheduling controller designed at every point on the trajectory. Therefore, it provides robust stability. TLC has been applied on fixed-wing aircraft loss-of-control prevention and recovery [18], tri-copter drone [19], fixed-wing aircraft [20] and omni-direction mobile robot [21]. It is noted that the vehicles in those applications are free of nonholonomic motin constriants. TLC is first applied to nonholonomic car-like ground vehicle for trajectory tracking in [17]. Figure 3 shows the 4-loop 3DOF TLC trajectory tracking controller. The dynamic pseudo-inverse of the corresponding EOMs is used to generate the nominal control signals, and the PI feedback controllers are used to stabilize the tracking error in each loop, where PI stands for Proportional-Integral controller. The guidance outer loop takes position command P_{com} from a trajectory generator and position measurement P_{sen} to calculate the velocity control V_{ctrl} for the guidance inner loop. The guidance inner loop takes velocity command $V_{com} = V_{nom} + V_{ctrl}$ from the the guidance outer loop and velocity measurement V_{sen} to calculate the force control F_{ctrl} for the guidance loop control allocation. The guidance allocation unit calculates the throttle command $\delta_{T,com}$ based on $F_{x,com} = F_{x,nom} + F_{x,ctrl}$ and the yaw command ψ_{com} as described below.

The car-like ground vehicle is subject to a nonholonomic constraint, for which lateral force cannot generate displacement. In other words, there are three variables, position $\begin{bmatrix} P_x & P_y \end{bmatrix}^T$ and orientation ψ need to be controlled, while there are only two actuators, throttle and steering servo. So in this paper, we use steering servo to control orientation and lateral position P˙y. Thus $F_{y,nom}$ and $F_{y,ctrl}$ are not used for lateral control. Instead the vehicle's rotational motion is used to correct the lateral position error. From Figure 2,

$$\psi = \chi - \beta \tag{1}$$

where β is vehicle sideslip angle. The nominal course angle χ_{nom} is defined as $\chi_{nom} = \tan^{-1}\frac{\dot{x}_{n,nom}}{\dot{y}_{n,nom}}$, where $\begin{bmatrix} \dot{x}_{n,nom} & \dot{y}_{n,nom} \end{bmatrix}^T$ is vehicle n-frame velocity. Since β makes the dynamic inverse unstable, it is excluded from the nominal dynamic inverse. So let $\beta_{nom} = 0$, then $\psi_{nom} = \chi_{nom}$. We will let the feedback controller deal with β. Define

$$\psi_{ctrl} = \beta_{ctrl} = \tan^{-1}\frac{v_{ctrl}}{u_{nom}} \tag{2}$$

where v_{ctrl} is the second channel of V_{ctrl} from the guidance outer loop. So $\psi_{com} = \psi_{nom} + \psi_{ctrl}$. The steering outer loop takes yaw command ψ_{com} and yaw measurement ψ_{sen} to calculate body

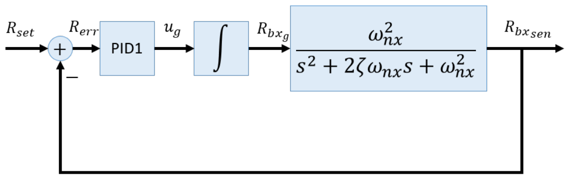
R_{set}
R_{err}
+
−
PID1
u_g
$\int$
R_{bx_g}
$\frac{\omega_{nx}^2}{s^2+2\zeta\omega_{nx}s+\omega_{nx}^2}$
$R_{bx_{sen}}$

Figure 4. Equivalent Closed-loop System for the Speed Guidance Law

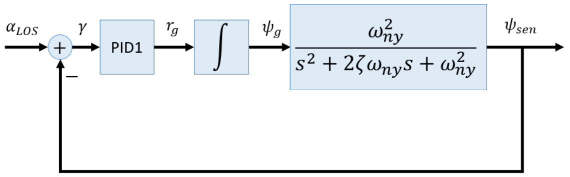
α_{LOS}
γ
+
−
PID1
r_g
$\int$
ψ_g
$\frac{\omega_{ny}^2}{s^2+2\zeta\omega_{ny}s+\omega_{ny}^2}$
ψ_{sen}

Figure 5. Equivalent Closed-loop System for the Heading Guidance Law

yaw rate command $r_{com} = r_{nom} + r_{ctrl}$. The Steering inner loop takes body yaw rate command r_{com} and yaw rate measurement r_{sen} to calculate the torque command $N_{m,com} = N_{m,nom} + N_{m,ctrl}$. And the steering allocation unit allocates the torque command to the steering actuator δ_S.

The overall close-loop position trajectory tracking controller is well decoupled into steering and position control in the b-frame at different bandwidth, with the following transfer functions:

$$\begin{aligned}\frac{R_{bx_{sen}}(s)}{R_{bx_g}(s)} &= \frac{\omega_{nx}^2}{s^2+2\zeta\omega_{nx}s+\omega_{nx}^2}\\ \frac{\psi_{sen}(s)}{\psi_g(s)} &= \frac{\omega_{ny}^2}{s^2+2\zeta\omega_{ny}s+\omega_{ny}^2}\end{aligned} \tag{3}$$

For the 1/6-scale Traxxas remotely-controlled (RC) model car in [17], which will also be used in this paper to demonstrate the LOS PPG guidance law, $\zeta = 0.7, \omega_{nx} = 3.0$ rad/sec, $\omega_{ny} = 5.5$ rad/sec.

LOS PPG Design

The LOS PPG algorithm can be used in many trajectory tracking applications, either for cooperative pursuit, such as vehicle platoon, or passive pursuit, like adaptive cruise control and lane tracking, or even adversarial pursuit, such as vehicle to vehicle chasing, like an unmanned law enforcement vehicle chasing a fugitive vehicle, as shown in Figure 6 . The vehicle to vehicle chasing case is used to discuss the LOS PPG design. Discussion of other applications of the LOS PPG will ensue.

In the present work, it is assumed that the pursuing vehicle's maneuverability is better than the target vehicle, and the road conditions are ideal. There are four objectives for LOS PPG trajectory generator: 1) acquire the LOS in the b-frame; 2) align the velocity vector with LOS; 3) keep a safety longitudinal distance between chasing vehicle and the target; 4) generate a nominal guidance trajectory P_g in Cartesian n-frame, which will be the input to the Trajectory Tracking Controller as P_{com} in Figure 3.

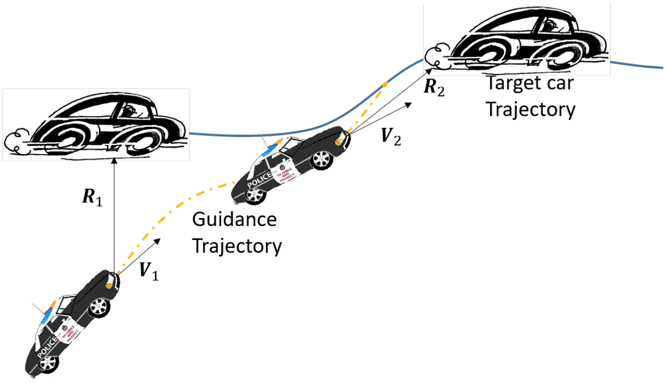

Figure 6. Diagram for Guidance Logic

Define the n-frame line-of-sight angle α_{LOS} as

$$\alpha_{LOS} = \tan^{-1}\frac{P_{y_{tgt}} - P_y}{P_{x_{tgt}} - P_x} \tag{4}$$

where $\left[P_{x_{tgt}} \;\; P_{y_{tgt}}\right]^T$ and $\left[P_x \;\; P_y\right]^T$ are target and pursuing vehicle's n-frame position respectively. Let

$$\mathbf{R} = \left[P_{x_{tgt}} - P_x,\, P_{y_{tgt}} - P_y\right]^T \tag{5}$$

be the range vector in the n-frame. The objective of LOS PPG is to align the velocity vector with the LOS, then the following equation is satified

$$\mathbf{V} \times \mathbf{R} = 0 \;\Rightarrow\; \alpha_{LOS} - \chi_g = 0 \tag{6}$$

where the subscript $_g$ stands for the guidance signals.

The LOS PPG trajectory generator is shown in Figure 7. It consists of four subsystems, one for each of the LOS PPG objectives, which will be described below. Tracing back to Figure 1, Figure 7 is subject to Guidance Trajectory Generator sub-blox and Figure 3 is subject to Trajectory Tracking Controller sub-blox.

Target Seeker

The Target Seeker subsystem is used to calculate the LOS in the b-frame. It is used to acquire the b-frame LOS angle γ with respect to the longitudinal axis x_b, and to determine the longitudinal down range distance R_{bx}, as shown in Figure 2.

The LOS in the n-frame is given in Eq (4), then b-frame LOS γ is given as $\gamma = \alpha_{LOS} - \psi_{sen}$, where ψ_{sen} is current instant navigation sensor sensed heading angle. The down range vector $\mathbf{R}_b$ in the b-frame is given by $\mathbf{R}_b = \begin{bmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{bmatrix}\mathbf{R}$. Let $R_b = \sqrt{\left(P_{x_{tgt}} - P_x\right)^2 + \left(P_{y_{tgt}} - P_y\right)^2}$ be the range between the pursuing vehicle and the target. $R_{bx} = R_b \cos\gamma$ and $R_{by} = R_b \sin\gamma$ are the range projection to the x_b and y_b respectively, where γ is the angle between line-of-sight and longitudinal axis x_b.

Heading Guidance

The Heading Guidance subsystem is used to align the velocity vector with LOS. As we said in the Technique Preliminary section, the nominal vehicle sideslip angle $\beta_{nom} = 0$ in the trajectory controller and it is dealt by the feedback controller. So in this paper, let $\beta_g = 0$ in the Guidance Trajectory Generator, then $\chi_g = \psi_g$ based on Eq (1). Then Eq (6) can be rewritten as

$$\gamma = \alpha_{LOS} - \psi_g = 0, \text{ as } t \mapsto \infty \tag{7}$$

Thus, the heading guidance objective is to exponentially stabilize $\gamma = 0$ using ψ_g as a virtual control, for which a Proportional-Integral-Derivative (PID) guidance law is employed

$$\begin{aligned} r_g &= K_{I1} \int_0^t \gamma(\tau)dt + K_{P1}\gamma + K_{D2}\dot{\gamma} \\ \psi_g &= \int_0^t r_g(\tau)d\tau \end{aligned} \tag{8}$$

The PID gains can be determined using Linear Time Invariant (LTI) controller design technique based on the equivalent closed loop system shown in Figure 5.

Speed Guidance

The Speed Guidance subsystem is used to keep a safety longitudinal distance between the pursuing vehicle and the target.

Define $R_{err} = R_{bx} - R_{set}$, where R_{set} is the safety longitudinal distance setting. In general, $R_{set} = \tau|V_x|$ is a velocity dependent parameter, where τ is an appropriate response time of the controller.

A second PID controller as shown in Figure is employed to exponentially stabilize $R_{err} \to 0$ as $t \to \infty$

$$u_g = K_{i2} \int_0^t \hat{R}_{err}(\tau)dt + K_{p2}\hat{R}_{err} + K_{d2}\dot{\hat{R}}_{err} \tag{9}$$

where $\hat{R}_{err} = \text{sat}_{R_{err,\max}}(R_{err})$, where $\text{sat}_a(x)$ is the saturation function defined by

$$\text{sat}_a(x) = \begin{cases} a, & x > a \\ x, & |x| \le a \quad , \ a > 0 \\ -a, & x < -a \end{cases} \tag{10}$$

The limit $R_{err,\max}$ is chosen to limit the maximum acceleration $\dot{u}_g$ due to large range errors, together with an integrator anti-wind up scheme.

Trajectory Synthesizer

The Trajectory Synthesizer subsystem is used to construct the nominal guidance trajectory P_g in the n-frame using the output of the Heading Guidance and Speed Guidance as follows

$$\begin{aligned} \dot{P}_{x_g} &= \hat{u}_g \cos\psi_g - v_g \sin\psi_g \\ \dot{P}_{y_g} &= \hat{u}_g \sin\psi_g + v_g \cos\psi_g \\ \hat{u}_g &= \text{sat}_{u_{\max}}(u_g) \\ P_{x_g} &= \int_0^t \dot{P}_{x_g}(\tau)d\tau \\ P_{y_g} &= \int_0^t \dot{P}_{y_g}(\tau)d\tau \end{aligned} \tag{11}$$

where v_g is vehicle lateral velocity in the b-frame. Since the side slip angle $\beta_g = 0$, then $v_g = 0$. The $u_{\max}$ is the maximum speed of the car taking into account of skidding prevention, which can be adaptively set in real time based on the operating and pavement conditions.

The LOS PPG algorithm developed above is for pursuing an uncooperative target without road constraint. This algorithm can also be used for cooperative pursuing, where the target broadcasts its coordinate P_{tgt} to the seeker. For neutral pursuit, such as lane keeping, the seeker will designate a virtual target in front of the vehicle on the center of the lane, see Figure 8, γ is then defined as the angle between the body frame x-axis x_b and the down range vector $\mathbf{R}_b$.

For vehicle pursuing with lane constraint, assume there are two targets, one is the real target, and the the other is the virtual target on the center of the lane, see Figure 9, where γ_1 is the angle between vehicle x-axis with real target down range vector $\mathbf{R}_{b1}$, and γ_2 is the angle between vehicle x-axis with virtual target down range vector $\mathbf{R}_{b2}$, as shown in the figure. If $\gamma_{err1} > \gamma_{err2}$, the vehicle chases the virtual target for the lane constraint, otherwise it chases the real target.

Simulation and Results

In order to verify the LOS PPG algorithm, it is implemented and simulated as MATLAB/SIMULINK. The vehicle model is

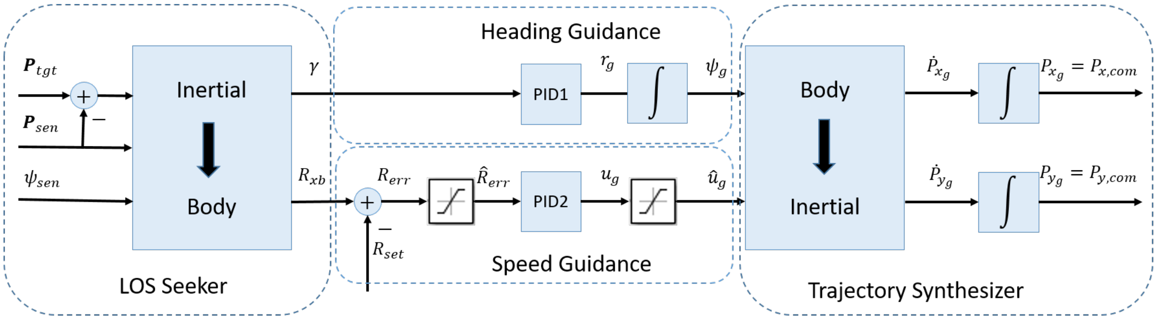

Figure 7. LOS PPG Trajectory Generator Architecture

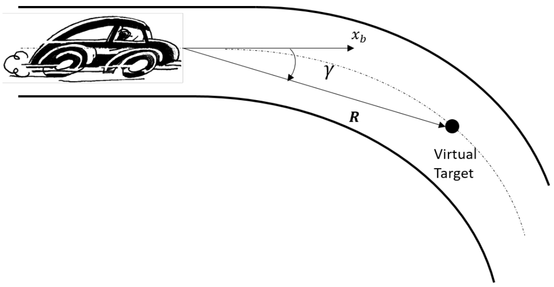

Figure 8. Lane Keeping Diagram $(R_{set} = 0)$

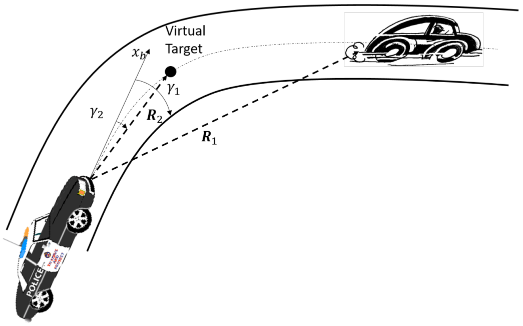

Figure 9. Vehicle Pursuing with Lane Constraint$(R_{set} = 0)$

Table 1. Vehicle Modeling Parameters for Traxxas RC vehicle

R_{eff}	Effective wheel radius	0.0725	m
L	Length of the longitudinal axle	0.7	m
w	Width of the vehicle track	0.35	m
m	Total mass	4.76	kg
I_{zz}	Moment of inertia about z-axis	0.0687	kgm^2
C_α	Total longitudinal tire stiffness	400	N/rad
C_{rr}	Total tire rolling resistance	0.02	
C_β	Total cornering stiffness	150	N/rad
J_m	Motor moment of inertia	$5.4e^{-5}$	kgm^2
B_m	Motor viscous friction constant	$0.5e^{-5}$	$Nm/(rad \cdot s)$
K_m	Motor KM constant	$5.15e^{-3}$	$N \cdot m/A$
R_a	Total armature resistance	0.2	Ω
N	Gear ratio	$\frac{1}{20}$	

Table 2. PID Parameter Setting

	K_p	K_i	K_d
Heading	0.4909	0.0067	1
Steering	0.15	0.0056	0

based on a 1/6-scaled Traxxas electric RC car. The model is implemented as a nonlinear 6DOF rigid-body model with 3 DOF motion constraints, as given in [17]. The vehicle parameters are given in Table 1. The baseline 3DOF trajectory tracking controller design and all controller parameters are also given in [17]. Based on the PPG tuning result, the parameters for the PID controller can be set as in Table 2.

Figure 10 to figure 18 show the MATLAB/SIMULINK simulation results. In the simulation, we set $R_{set} = 1.5$ m with the controller response time $\tau = 0.5$ s. Figure 10 is the 2-dimensional trajectory shown in the nframe. Figure 11 is the position tracking result. The target is initialized at 20 meters to the North of the origin with speed 4m/s towards the East along straight-S-straight trajectory to demonstrate the tracking performance of straight and turning performances. The controlled vehicle was initialized to accelerate from 0 m/s at the origin towards the North. The solid red line is the target trajectory, and the black dash line indicates the pure pursuit guidance generated trajectory and the blue dash-dot line is the chasing vehicle trajectory. Figure 12 shows the n-frame tracking error and the b-frame tracking error is shown in Figure 13. Figure 14 shows the body frame velocity tracking, which demonstrates the essential difference between

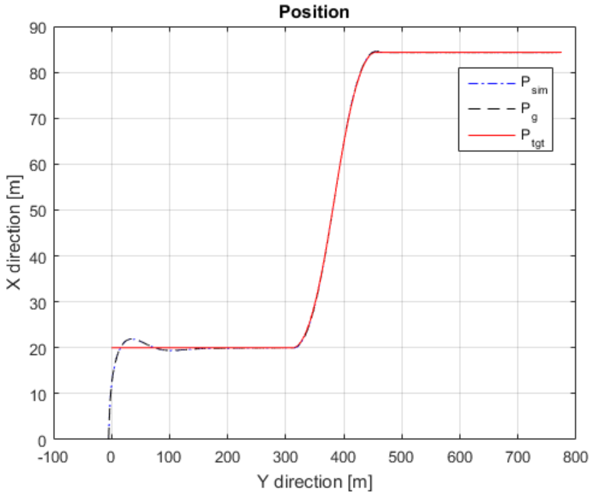

Figure 10. Trajectory Tracking for S-shape Motion Target

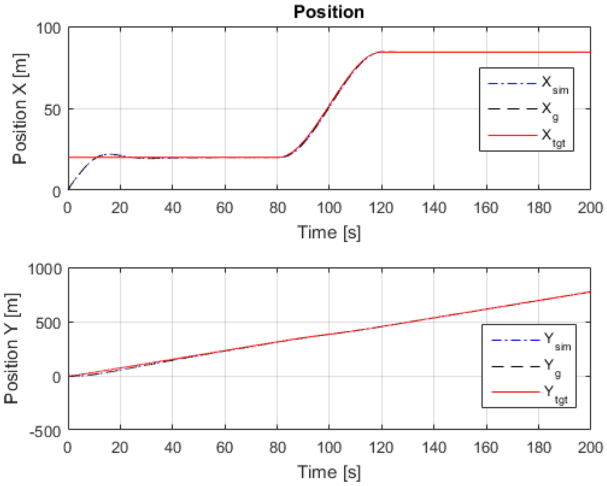

Figure 11. Position Tracking for S-shape Motion Target

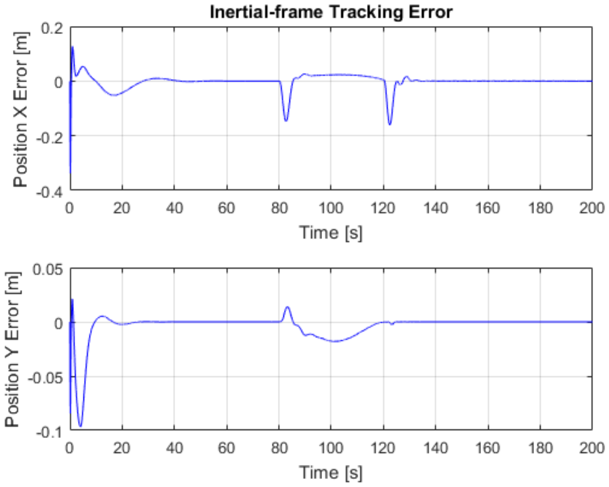

Figure 12. Tracking error in n-frame

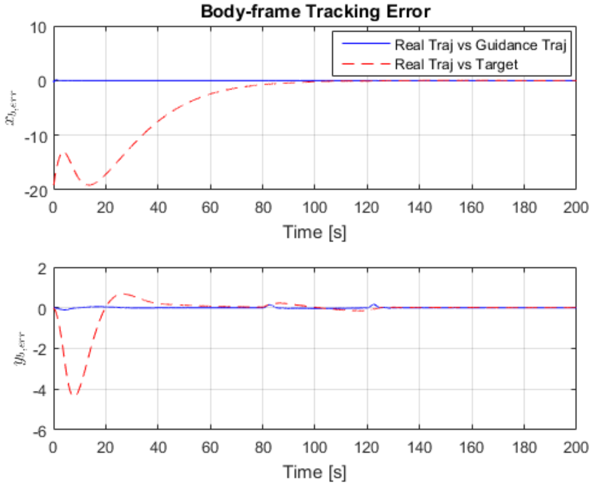

Figure 13. Tracking error in b-frame

the trajectory tracking and path-following. The longitudinal velocity tracking is smooth. The 1/6-scale vehicle start at 0 m/s, and acquired the target speed of 4 m/s, which is equivalent to 24 m/s, or 54 mph for a full-size car at around 60 s. The oscillatory transient in V_y is due to vehicle steering maneuver; side-slip angle β generates a lateral velocity component. Figure 15 and figure 16 show γ and R_{err} respectively. Both γ and R_{err} finally converge to 0, which fulfill our guidance objective. Figure 17 is heading angle ψ tracking with corresponding tracking error. Figure 18 shows the vehicle steering angle δ, overall the turning is smooth, but with some jitters which are due to the anomaly in calculating arctan function. Figure 19 is the sideslip angle β.

Conclusion and Future Work

A line-of-sight (LOS) pure pursuit guidance (PPG) design for 3DOF car-like non-holonomic ground vehicle has been presented in this paper. The LOS was decoupled into heading and speed guidance laws in the body frame to overcome the lateral non-holonomic constraint. A guidance trajectory is generated for the 3DOF inertial position trajectory controller instead of generating acceleration command. It is believed to be the first time LOS PPG has been employed for non-holonomic vehicle trajectory planning. The algorithm has been verified on MATLAB/SIMULINK based on a scaled electric car model with good result. In the near future, we will validate the algorithm on a model car, and expand the applications of the algorithm for lane

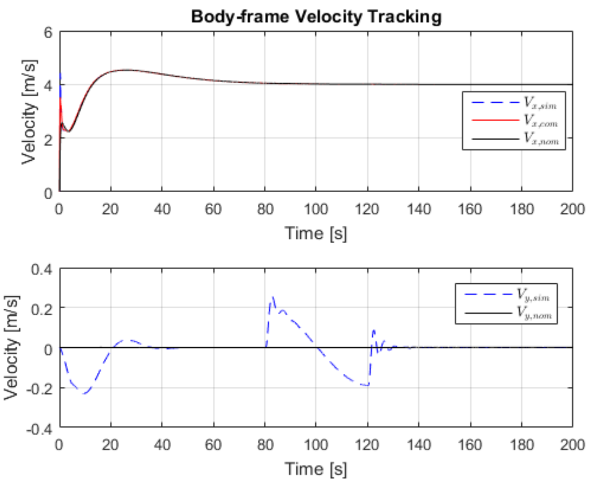

Figure 14. Body-frame Velocity Tracking

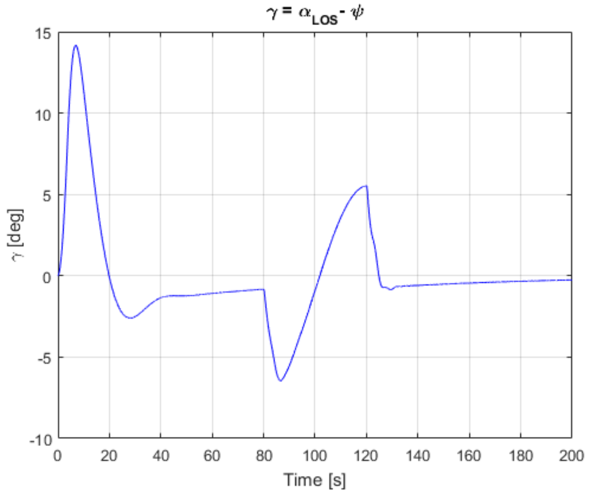

Figure 15. S-shape: $\gamma = \alpha_{LOS} - \psi$

chasing, parallel parking, and obstacle course trajectory planning.

REFERENCES

[1] National Highway Traffic Safety Administration. ”2015 motor vehicle crashes: overview.” Traffic safety facts research note 2016 (2016): 1-9.

[2] Paden, Brian, et al. ”A Survey of Motion Planning and Control Techniques for Self-driving Urban Vehicles.” IEEE Transactions on Intelligent Vehicles 1.1 (2016): 33-55.

[3] SAE On-Road Automated Vehicle Standards Committee. ”Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems.” (2014).

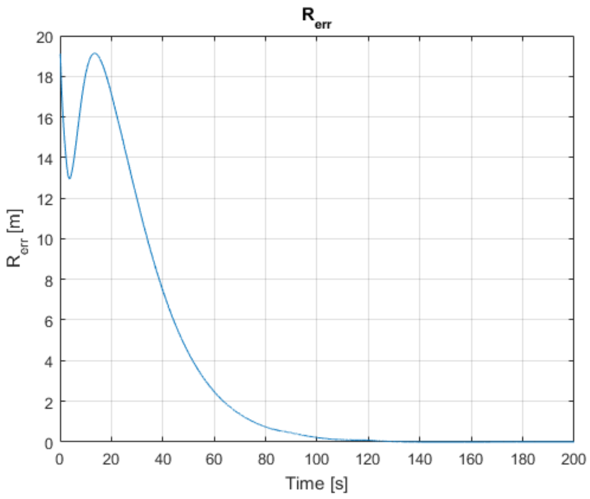

Figure 16. S-shape: Range Error R_{err}

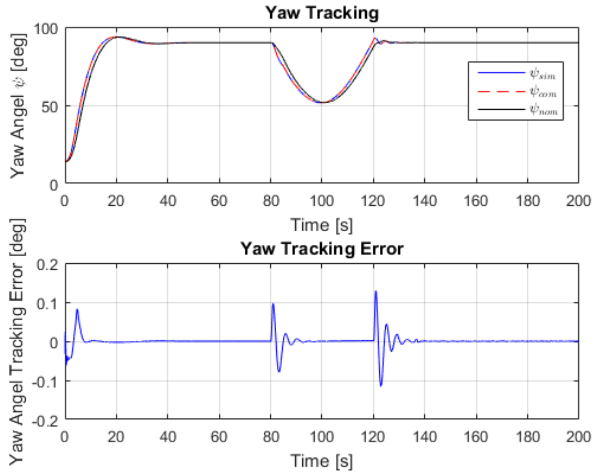

Figure 17. S-shape: Yaw Angle ψ

[4] Aguiar, A. Pedro, and Joao P. Hespanha. ”Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty.” IEEE Transactions on Automatic Control 52.8 (2007): 1362-1379.

[5] Amidi, Omead, and Chuck E. Thorpe. ”Integrated mobile robot control.” Fibers' 91, Boston, MA. International Society for Optics and Photonics, 1991.

[6] Wallace, Richard, et al. ”First Results in Robot Road-Following.” IJCAI. 1985.

[7] Giesbrecht, J., et al. ”Path tracking for unmanned ground vehicle navigation.” DRDC Suffield TM 224 (2005).

[8] Morales, Jess, et al. ”Pure-pursuit reactive path tracking for nonholonomic mobile robots with a 2D laser scanner.”

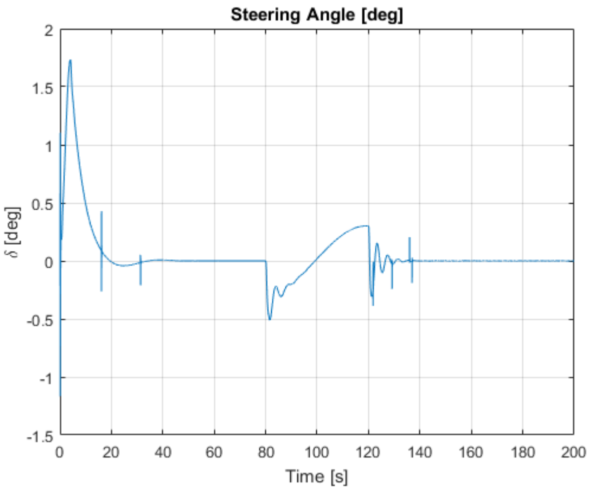

Figure 18. S-shape: Steering angle δ

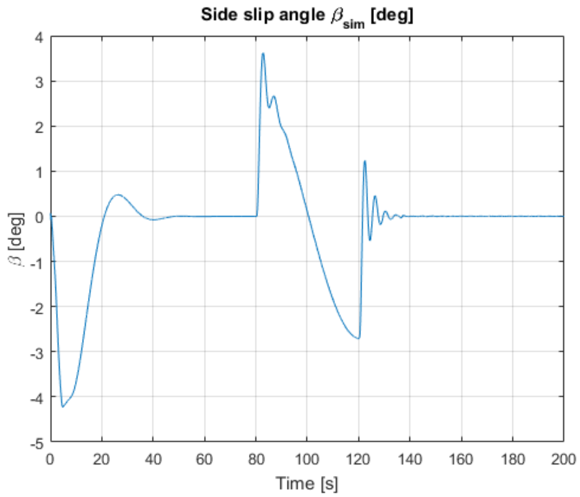

Figure 19. S-shape: Sideslip angle β

EURASIP Journal on Advances in Signal Processing 2009.

[9] Szepe, Tamas, and Samy FM Assal. ”Pure Pursuit Trajectory Tracking Approach: Comparison and Experimental Validation.” International Journal of Robotics and Automation 27.4 (2012): 355.

[10] Shneydor, Neryahu A. Missile guidance and pursuit: kinematics, dynamics and control. Elsevier, 1998.

[11] Garcia, Carlos E., David M. Prett, and Manfred Morari. ”Model predictive control: theory and practicea survey.” Automatica 25.3 (1989): 335-348.

[12] Morari, Manfred, and Jay H. Lee. ”Model predictive control: past, present and future.” Computers & Chemical Engineering 23.4 (1999): 667-682.

[13] Weiskircher, Thomas, and Beshah Ayalew. ”Predictive trajectory guidance for (semi-) autonomous vehicles in public traffic.” American Control Conference (ACC), 2015. IEEE, 2015.

[14] Consolini, Luca, Aurelio Piazzi, and Mario Tosques. ”A dynamic inversion based controller for path following of car-like vehicles.” IFAC Proceedings Volumes 35.1 (2002): 49-54.

[15] Kanayama, Yutaka, et al. ”A stable tracking control method for an autonomous mobile robot.” Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on. IEEE, 1990.

[16] d’Andra-Novel, Brigitte, Gianni Campion, and Georges Bastin. ”Control of nonholonomic wheeled mobile robots by state feedback linearization.” The International journal of robotics research 14.6 (1995): 543-559.

[17] Chen, Yuanyan and J.Jim Zhu. ”Car-like Ground Vehicle Trajectory Tracking by using Trajectory Linearization Control”, 2017 ASME Dynamic Systems and Control Conference (Accepted).

[18] Zhao, Yue, and J. Jim Zhu. ”Aircraft loss-of-control autonomous recovery: Mission trajectory tracking restoration.” Unmanned Aircraft Systems (ICUAS), 2016 International Conference on. IEEE, 2016.

[19] Huang, Rui, et al. ”Guidance, navigation and control system design for a tri-propeller VTOL UAV.” AIAA Guidance, Navigation and Control Conference and Exhibit, Hilton Head, South Carolina. 2007.

[20] Adami, Tony M., and J. Jim Zhu. ”6DOF flight control of fixed-wing aircraft by trajectory linearization.” Proceedings of the American Control Conference. IEEE, 2011.

[21] Liu, Yong, et al. ”Omni-directional mobile robot controller based on trajectory linearization.” Robotics and Autonomous Systems 56.5 (2008): 461-479.