RST Longitudinal Control Design for Smart Car Platform in Autonomous Driving

Duc Lich LUU[1], Ciprian LUPU[2], Van Ga BUI[1], Nguyen Viet HUNG[3]*

[1] Faculty of Transportation Mechanical Engineering, The University of Da Nang – University of Science and Technology, 54 Nguyen Luong Bang, 550000, Da Nang, Vietnam
ldlich@dut.udn.vn, bvga@dut.udn.vn

[2] Department of Automatic Control and Systems Engineering, National University of Science and Technology Politehnica Bucharest, 313 Splaiul Independenței, 060042, Bucharest, Romania
ciprian.lupu@upb.ro

[3] International Training and Cooperation Institute, East Asia University of Technology, 100000, Hanoi, Vietnam
hungnv@eaut.edu.vn (**Corresponding author*)

Abstract: As one of the most important Advanced Driving Assistant Systems (ADASs), Adaptive Cruise Control (ACC) system is the initial development stage of intelligent driving technology, which is extended from the Cruise Control (CC) system. Leading car using CC system is moving with a desired velocity, and a host car using ACC system approaches the preceding vehicle and it maintains the actual distance close to the desired distance, based on both the constant time headway policy and the constant spacing policy. This study presents a design methodology for both CC and ACC systems, in the discrete time system, by using RST (reference signal tracking) polynomial Controller. Digital Control Design by the polynomial method, as RST polynomial, has outstanding advantages such as: simplicity, flexibility and real-time applicability in the discrete time system. A new smart car platform, which has been introduced in a previous work, is embedded and tested to evaluate the performance and effectiveness of the RST structure in the discrete time system, for both CC and ACC systems. Simulation and testing results show that the RST algorithms proposed in this study have good applicability.

Keywords: ADASs, RST controller, Adaptive cruise control, Automotive electronics, Real-time systems.

1. Introduction

Automated vehicle control systems are an important part of autonomous vehicles, representing the core of Intelligent Vehicle Systems (IVSs) (Dokur & Katkoori, 2023; Rajabli et al., 2021).

Recently, Advanced Driving Assistant System (ADAS) has become the starting system development stage of IVSs technology (Tiganasu et al., 2021, Wang et al., 2024). It can use a sophisticated control and sensing elements to collect relevant data and realize the possibility of automatic vehicle control (Hu et al., 2023; Dey et al., 2015). As an ADAS system, Adaptive Cruise Control feature has been integrated into the passenger cars. It can automatically adjust the velocity of the controlled car to keep a desired safe spacing (Yang et al., 2024; Ma et al., 2022), the purposes of this system being to increase drivers' safety, decrease accident rates, and improve traffic flow (Duffney et al., 2022; Sakhdari & Azad, 2018).

Fully automatic Cruise Control systems are a crucial part of the new architecture of modern highway systems, which tend to increase traffic flow (Dawood et al., 2018). The function of CC control system can be used to take over the control of the car according to the desired velocity set by the driver, instead of the driver regularly adjusting the throttle and brake pedal. Therefore, this function has the added benefit of reducing driver fatigue during a long road trip (Sivaji & Sailaja, 2013; Zhou et al., 2022). ACC system is extended from the CC system, as a more advanced version of it (Pan et al., 2022). In 1990s, this system has been applied for luxury vehicles in many companies such as Mercedes-Benz, Audi, BMW, etc.

The commonly used methods for the design of ACC algorithms include model predictive control (Al-Gabalawy et al., 2021), sliding mode control (Wu et al., 2019) and Fuzzy PI control (Maji et al., 2015). Fuzzy logic controller has been used for automatic ACC system by (Tsai et al., 2010), while an adaptive neuro-fuzzy predictive control has proposed in (Avdagic et al., 2019).

For real-time testing for smart car area, mobile robot platform has been of interest for some authors (Yue, Guo & Yuan, 2017; Trudgen et al., 2018) have proposed the application of robot platooning for the ACC with different controllers, such as predictive controller and sliding mode controller, and have taken into account the noise of both the devices and the motors. The area of smart cars has been developed and tested in several other papers (Lupu et al., 2018) with the purpose of avoiding collisions and spotting people and fires, while tracking the lane system, at the same time.

https://doi.org/10.24846/v34i1y202508

However, one of the major problems of using these algorithms is the large and complex computation amount. Thus, it is difficult to implement them in real-time application or in discrete time system.

As far as known, there are very few scientific works on RST algorithms applied to autonomous cars, especially ADAS systems. In this study, RST algorithm is proposed to design both the CC and the ACC system. Digital Control Design by the polynomial method, such as RST algorithm, has two outstanding advantages: simplicity and real-time applicability. A RST control approach has been introduced by some scholars as (Lupu et al., 2014; Ayadi et al., 2008).

This paper focuses on implementing digital control design by the polynomial method in the discrete time system for ADAS systems. The leader car is designed by the CC system with the reference of the closed-loop (CL) system, in order to obtain the desired velocity, using a RST controller. The host car is designed by the ACC system with the reference of the CL system, in order to obtain the desired distance, using a RST controller. The purpose is to follow the leading car at the desired distance, based on both the Constant Time Headway Policy (CTHP) and the Constant Spacing Policy (CSP).

Next, the experiment verifies the performances of two smart cars consisting of a leading (CC system) and a host smart car (ACC system), with the constant spacing policy.

The present study is structured as follows. Section 2 considers the modelling of the cars. Then, CC system design for the leading car is presented in Section 3. In Section 4, ACC system design for host car is presented. Section 5 describes the hardware structure of the present approach. In Section 6, the illustrative examples and the testing results are provided through the performance analysis and, finally, Section 7 offers the conclusions of this study.

2. Modelling of Cars

Considering two cars which run on a single lane, namely the leading car and the host car. Assume that the operation of the host car installed with the ACC system looks at a single driving car, as it can be seen in Figure 1, and each car is equipped with sensors meant to measure the distance and the relative velocity to the leading car.

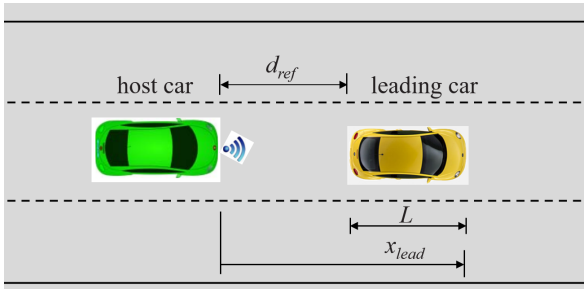

Figure 1. Two cars following each other on a single lane (Luu et al., 2023)

According to Newton's second law, the balance of the forces acting on the vehicle longitudinal axis is described as follows (Rajamani, 2011; Ulsoy et al., 2012):

$$m_i\ddot{x}_i(t) = F(t) - mg\sin\delta - fmg\cos\delta \\ -0.5\rho AC_{di}v^2(t) \tag{1}$$

where, the position, velocity, and acceleration in longitudinal axis of the leader car at time instant t are indicated by $x(t)$, $v(t)$, $a(t)$, respectively. m is the car mass, F is the traction force provided by the engine, the resistances of car includes air resistance, road friction, ramp resistance, in which A is the cross-sectional area of car, C_{di} is the drag coefficient, ρ is the air density, f is the road friction coefficient, δ is the road slope, g is the gravitational acceleration. The actual acceleration variation of the car may be indicated as:

$$\dot{a}_i(t) = \frac{1}{m\varsigma}\left[F_u(t) - F(t)\right] \tag{2}$$

where the time constant of the lag in tracking any desired acceleration command of the car is set to ς. Assume that $\varsigma \le \varsigma_0$ and ς_0 is an upper bound on the time lag constant, thus it can be chosen as 0.2s.

Replace formula (1) into formula (2):

$$\dot{a}_i(t) = \frac{1}{m\varsigma}\begin{pmatrix} F_u(t) - ma(t) - mg\sin\delta \\ -fmg\cos\delta - 0.5\rho AC_{di}v^2(t) \end{pmatrix} \tag{3}$$

The expected traction force is given as:

$$F_u(t) = mu(t) + mg\sin\delta + fmg\cos\delta \\ +0.5\rho AC_{di}v^2(t) \tag{4}$$

Now, replace formula (3) into formula (4), the longitudinal dynamics of each car in the platoon can be written as:

$$\varsigma\dot{a}(t) + a(t) = u(t) \tag{5}$$

Then, the dynamic equation of longitudinal dynamic model of the car is of the form:

$$\begin{cases} \dot{x}(t) = v(t) \\ \ddot{x}(t) = a(t) \\ \varsigma\dddot{x}(t) + \ddot{x}(t) = u(t) \end{cases} \quad (6)$$

From formula (6), the transfer function between the commanded acceleration and the velocity or the longitudinal position of the car model is given by:

$$G(s) = \frac{V(s)}{U(s)} = \frac{1}{s(\varsigma s + 1)} \quad (7)$$

or

$$\widetilde{G}(s) = \frac{X(s)}{U(s)} = \frac{1}{s^2(\varsigma s + 1)} \quad (8)$$

3. CC System Design for Leading Car

The leading car is designed by the CC system meant to adjust the velocity of car which is appropriate for the CL system. Based on formula (7), the transfer function of the car model $G(z^{-1})$ in the discrete time system for the digital system has the following general form (Landau & Zito, 2006; Popescu et al., 2017):

$$G(z^{-1}) = \frac{A(z^{-1})}{B(z^{-1})} \quad (9)$$

where, the polynomials $A(z^{-1})$ and $B(z^{-1})$ are the car model of CC system that is determined by the following expressions:

$$\begin{cases} A(z^{-1}) = 1 + \sum_{j=1}^{p_a=2} a_j.z^{-j} = 1 + a_1 z^{-1} + a_2 z^{-2} \\ B(z^{-1}) = 1 + \sum_{j=1}^{p_b=2} b_j.z^{-j} = b_0 + b_1 z^{-1} + b_2 z^{-2} \end{cases} \quad (10)$$

Figure 2 represents the CC structure of the leading car which is implemented by the RST controller based on the longitudinal dynamic model of the car formula (9). RST controller is described by:

$$\begin{aligned} u(k) &= \begin{bmatrix} -\frac{R(z^{-1})}{S(z^{-1})} & \frac{T(z^{-1})}{S(z^{-1})} \end{bmatrix} \begin{bmatrix} v(k) \\ v_{ref}(k) \end{bmatrix} \\ &= -\frac{R(z^{-1})}{S(z^{-1})} v(k) + \frac{T(z^{-1})}{S(z^{-1})} v_{ref}(k) \end{aligned} \quad (11)$$

with z^{-1} being the transfer function variable in discrete system, where $v_{ref}(k)$ is the desired velocity set by the human; $u(k)$ is the desired acceleration for car (control signal); $v(k)$ is the measured output signal of the physical system (velocity of car); and $R(z^{-1})$, $T(z^{-1})$, $S(z^{-1})$ polynomials are the weighting regulation block, the sensitivity block, and the tracking block, respectively. These blocks are given by:

$$\begin{aligned} R(z^{-1}) &= \sum_{j=0}^{p_r} r_j.z^{-j} = r_0 + r_1 z^{-1}, && \forall p_r \in \mathbb{N} \\ T(z^{-1}) &= \sum_{j=0}^{p_t} t_j.z^{-j} = t_0, && \forall p_t \in \mathbb{N} \\ \frac{1}{S(z^{-1})} &= \frac{1}{\sum_{j=0}^{p_s} s_j.z^{-j}} = \frac{1}{s_0 + s_1 z^{-1}}, && \forall p_s \in \mathbb{N} \end{aligned} \quad (12)$$

The control algorithm in formula (11) can be rewritten as follows:

$$u(k) = \frac{1}{s_0}\begin{bmatrix} T(z^{-1})v_{\text{ref}}(k) - r_0 v(k) - s_1 u(k-1) \\ -r_1 v(k-1) \end{bmatrix} \quad (13)$$

For the constraint condition, it is more suitable with the driver's operation. The constraints are defined as: $u_{\min}(k) \le u(k) \le u_{\max}(k)$, where $u_{\min}(k) < 0$ and $u_{\max}(k) > 0$ are bounds of control input for the leading car.

The CL transfer function between the desired velocity $v_{ref}(k)$ and the measured output signal

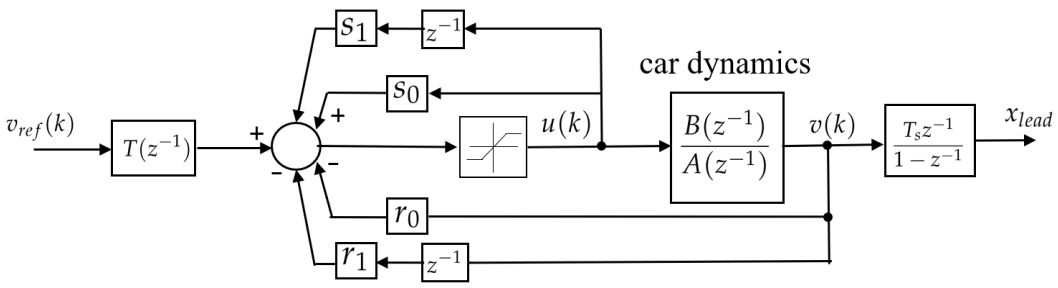

Figure 2. Digital controller of CC system for the leading car

$v(k)$, using a digital controller canonical structure, has the expression:

$$H_{CL}(z^{-1}) = \frac{B(z^{-1})T(z^{-1})}{A(z^{-1})S(z^{-1}) + B(z^{-1})R(z^{-1})} \quad (14)$$

In the present model the characteristic polynomial, $P(z^{-1})$ has the following order:

$$P(z^{-1}) = 1 + p_1 z^{-1} + p_2 z^{-2} \quad (15)$$

The characteristic polynomial $P(z^{-1})$ has the following form, as per formula (14):

$$P(z^{-1}) = A(z^{-1})S(z^{-1}) + B(z^{-1})R(z^{-1}) \quad (16)$$

The input/output transfer of the CL system is described by:

$$v(k) = \frac{B(z^{-1})T(z^{-1})}{P(z^{-1})} v_{\text{ref}}(k) \quad (17)$$

The polynomials $R(z^{-1})$ and $S(z^{-1})$ are determined through the following matrix equation:

$$\boldsymbol{X} = \boldsymbol{M}^{-1}\boldsymbol{p} \quad (18)$$

where:

$$\boldsymbol{X}^T = \begin{bmatrix} s_0 & s_1 & r_0 & r_1 \end{bmatrix}, \quad \boldsymbol{p}^T = \begin{bmatrix} 1 & p_1 & p_2 & 0 \end{bmatrix},$$

$$\boldsymbol{M} = \begin{bmatrix} a_0 & 0 & 0 & 0 \\ a_1 & a_0 & b_2 & 0 \\ a_2 & a_1 & b_2 & b_1 \\ 0 & a_2 & 0 & b_2 \end{bmatrix}$$

Usually, to calculate the polynomial controller, a second-degree transfer function is expressed as:

$$H_r(s) = \frac{\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2} v_{\text{ref}}(k) \quad (19)$$

where ζ is the damping coefficient and ω_0 is the natural frequency and the polynomials p_1, p_2 are determined as (Landau & Zito, 2006):

$$\begin{aligned} p_1 &= -2e^{-\zeta\omega_0 T_s}\cos(\omega_0\sqrt{1-\zeta^2}Ts \\ p_2 &= -e^{-2\zeta\omega_0 T_s} \end{aligned} \quad (20)$$

Based on the design method presented in (Stefanoiu et al., 2016), $T(z^{-1})$ polynomial coefficients are determined using the following formula:

$$T(z^{-1}) = \hat{P}(z^{-1})B(z^{-1}) \quad (21)$$

where $\hat{P}(z^{-1})$ is another unknown polynomial.

4. ACC System Design for Host Car

The ACC system is one of the applications of the longitudinal control of the car. Using the ACC system of the host car means maintaining the same velocity with the leading car, while also maintaining the desired distance value with respect to the leading car. Host car allows following its preceding car with the desired spacing defined by the spacing policies.

The structure of discrete time car model (7) used for digital controller design has the following general form (Landau & Zito, 2006; Popescu et al., 2017):

$$\tilde{G}(z^{-1}) = \frac{\tilde{A}(z^{-1})}{\tilde{B}(z^{-1})} \quad (22)$$

where the polynomials $\tilde{A}(z^{-1}), \tilde{B}(z^{-1})$ represent the car model of ACC system using the complex mathematical model determined as:

$$\begin{cases} \tilde{A}(z^{-1}) = 1 + \sum_{j=1}^{\tilde{p}_a=2} \tilde{a}_j . z^{-j} = 1 + \tilde{a}_1 z^{-1} + \tilde{a}_2 z^{-2} \\ \tilde{B}(z^{-1}) = 1 + \sum_{j=1}^{\tilde{p}_b=2} \tilde{b}_j . z^{-j} = \tilde{b}_0 + \tilde{b}_1 z^{-1} + \tilde{b}_2 z^{-2} \end{cases} \quad (23)$$

The ACC system structure of the host car which is implemented by the RST controller is illustrated in Figure 3. Clearly, RST controller is provided by:

$$\begin{aligned} \tilde{u}(k) &= \begin{bmatrix} -\frac{\tilde{R}(z^{-1})}{\tilde{S}(z^{-1})} & \frac{\tilde{T}(z^{-1})}{\tilde{S}(z^{-1})} \end{bmatrix} \begin{bmatrix} d(k) \\ d_{ref}(k) \end{bmatrix} \\ &= -\frac{\tilde{R}(z^{-1})}{\tilde{S}(z^{-1})} d(k) + \frac{\tilde{T}(z^{-1})}{\tilde{S}(z^{-1})} d_{ref}(k) \end{aligned} \quad (24)$$

In the ACC system is the desired distance defined by the CTHP and the CSP; $\tilde{u}(k)$ is the desired acceleration for host car (control signal); and L is the car length.

$$\begin{cases} d_{\text{ref}}(k) = c_0 + t_k v_{\text{host}}(k) \quad (m) \quad \text{for CTHP} \\ d_{\text{ref}}(k) = l_0(k) \quad (m) \quad \text{for CSP} \end{cases} \quad (25)$$

where the forward velocity $v_{\text{host}}(k)$ is weighted with the headway time t_k. c_0, t_k denote the distance to be kept at zero velocity, and the time headway, respectively.

$d(k)$ is the actual distance between the host car and the leading car. Iy is measured using radar sensor mounted on the front bumper of each car and it is defined as:

$$d_{\text{ref}}(k) = x_{lead}(k) - x_{\text{host}} - L \quad (26)$$

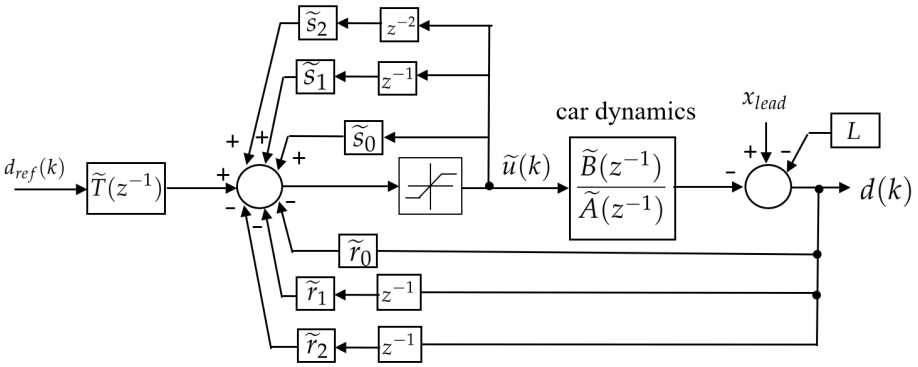

Figure 3. Digital controller of ACC system for host car

The polynomials $\tilde{R}(z^{-1})$, $\tilde{T}(z^{-1})$ and $\tilde{S}(z^{-1})$ of the proposed RST digital feedback controller are defined as below:

$$\begin{aligned}
&\tilde{R}(z^{-1}) = \sum_{j=0}^{\tilde{p}_r} \tilde{r}_j . z^{-j} = \tilde{r}_0 + \tilde{r}_1 z^{-1} + \tilde{r}_2 z^{-2}, \forall \tilde{p}_r \in \mathbb{N} \\
&\tilde{T}(z^{-1}) = \sum_{j=0}^{\tilde{p}_t} \tilde{t}_j . z^{-j} = \tilde{t}_0, \qquad \forall \tilde{p}_t \in \mathbb{N} \\
&\frac{1}{\tilde{S}(z^{-1})} = \frac{1}{\sum_{j=0}^{\tilde{p}_s} \tilde{s}_j . z^{-j}} = \frac{1}{\tilde{s}_0 + \tilde{s}_1 z^{-1} + \tilde{s}_2 z^{-2}}, \forall \tilde{p}_s \in \mathbb{N}
\end{aligned} \tag{27}$$

Taking into account the expression of $\tilde{S}(z^{-1})$, the control signal $\tilde{u}(z^{-1})$ is computed on the basis of formula (24), by means of the formula:

$$\begin{aligned}
\tilde{u}(k) = \frac{1}{\tilde{s}_0}\Big[&\tilde{T}(z^{-1}) d_{\text{ref}}(k) - \tilde{r}_0 d(k) - \tilde{r}_1 d(k-1) \\
&- \tilde{r}_2 d(k-2) - \tilde{s}_1 \tilde{u}(k-1) - \tilde{s}_2 \tilde{u}(k-2)\Big]
\end{aligned} \tag{28}$$

The constraint condition for the host car is the following: $\tilde{u}_{\min}(k) \le \tilde{u}(k) \le \tilde{u}_{\max}(k)$ where $\tilde{u}_{\min}(k) < 0$ and $\tilde{u}_{\max}(k) > 0$ are bounds of control input for the host car.

In this case, the transfer function in closed-loop is given by:

$$\begin{aligned}
\tilde{H}_{CL}(z^{-1}) &= \frac{\tilde{B}(z^{-1})\tilde{T}(z^{-1})}{\tilde{A}(z^{-1})\tilde{S}(z^{-1}) + \tilde{B}(z^{-1})\tilde{R}(z^{-1})} \\
&= \frac{\tilde{B}(z^{-1})\tilde{T}(z^{-1})}{\tilde{P}(z^{-1})}
\end{aligned} \tag{29}$$

The input/output transfer of the CL system is described by:

$$\begin{aligned}
d(k) &= \frac{\tilde{B}(z^{-1})\tilde{T}(z^{-1})}{\tilde{P}(z^{-1})} d_{\text{ref}}(k) \\
&+ \frac{\tilde{A}(z^{-1})\tilde{S}(z^{-1})}{\tilde{P}(z^{-1})}\big(-x_{lead}(k) + L\big)
\end{aligned} \tag{30}$$

Algorithm 1. RST scheme of the host car

Input: set the structural index of the model: the system parameters c_0, t_k, L, l_0, sampling time T_s, desired setpoint $d_{ref}(k)$, the constraint sets $\tilde{u}(k)$.

Output: actual distance $d(k)$

Start loop to compute:

1: Compute discrete time car model in order to achieve $\tilde{A}(z^{-1})$, $\tilde{B}(z^{-1})$

2: Solve (33) to compute $\tilde{R}(z^{-1})$ and $\tilde{S}(z^{-1})$ polynomials of the controller.

3: Solve (21) to obtain $\tilde{T}(z^{-1})$ polynomial

4: Measure output signal

$$\begin{aligned}
d(k) &= \frac{\tilde{B}(z^{-1})\tilde{T}(z^{-1})}{\tilde{P}(z^{-1})} d_{\text{ref}}(k) \\
&+ \frac{\tilde{A}(z^{-1})\tilde{S}(z^{-1})}{\tilde{P}(z^{-1})}\big(-x_{lead}(k) + L\big)
\end{aligned}$$

5: Generate control signal

$$\begin{aligned}
\tilde{u}(k) = \frac{1}{\tilde{s}_0}\Big[&\tilde{T}(z^{-1}) d_{\text{ref}}(k) - \tilde{r}_0 d(k) - \tilde{r}_1 d(k-1) \\
&- \tilde{r}_2 d(k-2) - \tilde{s}_1 \tilde{u}(k-1) - \tilde{s}_2 \tilde{u}(k-2)\Big]
\end{aligned}$$

6: Update $k \leftarrow k+1$ and go to step 1.

The calculation of the polynomials $\tilde{R}(z^{-1})$ and $\tilde{S}(z^{-1})$ is carried out starting from the choice of the polynomial $\tilde{P}(z^{-1})$:

$$\begin{aligned}
\tilde{P}(z^{-1}) &= \tilde{A}(z^{-1})\tilde{S}(z^{-1}) + \tilde{B}(z^{-1})\tilde{R}(z^{-1}) \\
&= 1 + \tilde{p}_1 z^{-1} + \tilde{p}_2 z^{-2}
\end{aligned} \tag{31}$$

where the polynomials $\tilde{p}_1, \tilde{p}_2$ are determined as (Landau & Zito, 2006):

$$\begin{aligned}
\tilde{p}_1 &= -2e^{-\zeta\omega_0 T_s \cos(\omega_0 \sqrt{1-\zeta^2} T_s} \\
\tilde{p}_2 &= -e^{-2\zeta\omega_0 T_s}
\end{aligned} \tag{32}$$

The coefficients of $\tilde{R}(z^{-1})$ and $\tilde{S}(z^{-1})$ contained in vector $\tilde{X}$ are given by:

$$\tilde{\boldsymbol{X}} = \tilde{\boldsymbol{M}}^{-1}\tilde{\boldsymbol{p}} \tag{33}$$

where:

$$\tilde{\boldsymbol{X}}^T = \begin{bmatrix} \tilde{s}_0 & \tilde{s}_1 & \tilde{s}_2 & \tilde{r}_0 & \tilde{r}_1 & \tilde{r}_2 \end{bmatrix},$$

$$\tilde{\boldsymbol{p}}^T = \begin{bmatrix} 1 & \tilde{p}_1 & \tilde{p}_2 & 0 & 0 & 0 \end{bmatrix}$$

$$\tilde{\boldsymbol{M}} = \begin{bmatrix} \tilde{a}_0 & 0 & 0 & 0 & 0 & 0 \\ \tilde{a}_1 & \tilde{a}_1 & 0 & \tilde{b}_1 & 0 & 0 \\ \tilde{a}_2 & \tilde{a}_1 & \tilde{a}_0 & \tilde{b}_1 & \tilde{b}_1 & 0 \\ \tilde{a}_3 & \tilde{a}_2 & \tilde{a}_1 & \tilde{b}_3 & \tilde{b}_2 & \tilde{b}_1 \\ 0 & \tilde{a}_3 & \tilde{a}_2 & 0 & \tilde{b}_3 & \tilde{b}_2 \\ 0 & 0 & \tilde{a}_3 & 0 & 0 & \tilde{b}_3 \end{bmatrix}$$

5. Hardware Structure Description

For a real-time application in the autonomous car area, a new smart car platform was made in (Luu et al., 2019). The experiment was performed as an approach technique of longitudinal control for autonomous car. This platform has several advantages such as simplicity and real-time applicability.

Since the smart cars are moved within a very small space, it is not practical to measure the distance between them by using GPS or another device. The infrared sensor is employed instead. Each device is tested in real time separately. For obtaining more details, several sensors are tested, as illustrated in Figure.4.

The smart car platform, known as "Arduino cars", can be seen from Figure 5, it considers one leading car and one host car. For the smart car platform, the velocity is measured by using signals from the wheel encoders on all wheels. Computations are needed, in order to determine the exact wheel velocity. In this way, smart cars will be able to keep a fixed distance between them. The micro-controller commands the motors through a motor driver controlled by pulse width modulation (*PWM*) signals.

This prototype smart car does not contain the complex automobile dynamics, it is static in the laboratory, there is no significant airflow around the car and the rolling resistance is based entirely on different phenomena. Therefore, an experimental method is proposed, namely the transfer function of the model, in order to identify the longitudinal dynamics in smart car platform.

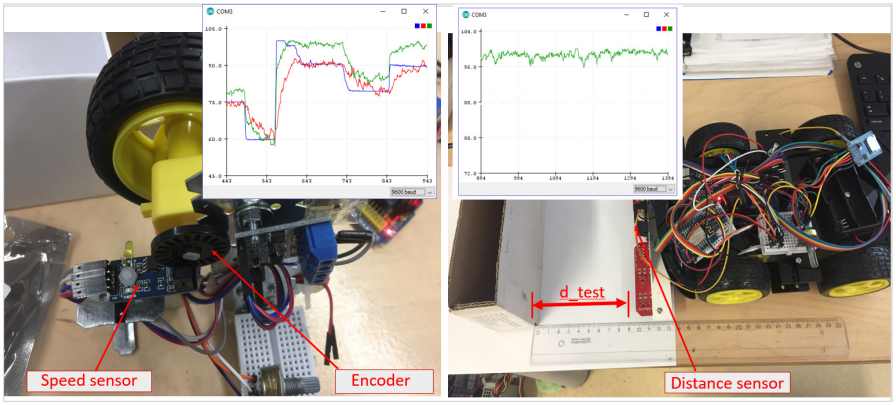

Figure 4. Testing infrared speed sensor and distance sensor (Viet Hung et al., 2024)

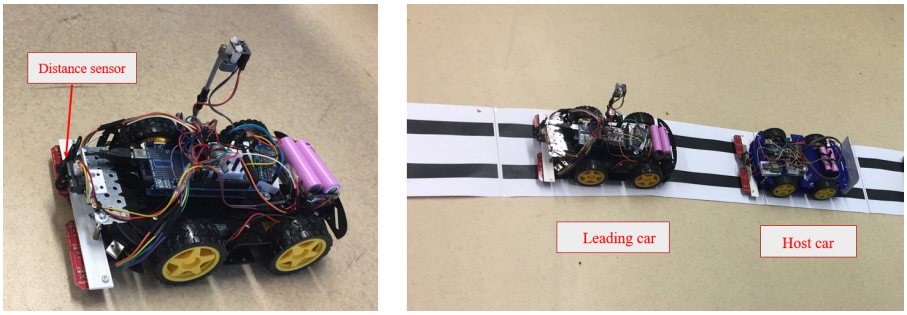

Figure 5. Experiments with the smart car platform in the laboratory (Luu et al., 2023)

6. Illustrative Example

6.1 Numerical Results

Before the start of the numerical example, the parameters of the car were established as follows: $\varsigma = 0.2s, L = 5(m)$. The system constraints are as follows:

$$|u(k)| \le 2.5(\frac{m}{s^2}),\ |\tilde{u}(k)| \le 2.5(\frac{m}{s^2}).$$

The choice of the polynomials R, S and T allows solving both the regulation and the tracking problem. By using a sampling period of $T_s = 0.1$ second and model parameters from (10) and (23), the controllers can be computed based on method proposed in (Chiriac, 2021; Popescu et al., 2017):

$$\begin{aligned} A(z^{-1}) &= 1 - 1.6065z^{-1} + 0.6065z^{-2} \\ B(z^{-1}) &= 0.0213z^{-1} + 0.018z^{-2} \\ R(z^{-1}) &= -0.0506 + 1.0316z^{-1} \\ T(z^{-1}) &= 0.9852 \\ S(z^{-1}) &= 1 - 0.0307z^{-1} \end{aligned} \quad (34)$$

and, for the host car

$$\begin{aligned} \tilde{A}(z^{-1}) &= 1 - 2.6065z^{-1} + 2.213z^{-2} - 0.6065z^{-3} \\ \tilde{B}(z^{-1}) &= 0.00073877z^{-1} + 0.0026z^{-2} \\ &+0.0005755z^{-3} \\ \tilde{R}(z^{-1}) &= 249.4258 - 387.5863z^{-1} + 148.0126z^{-2} \\ \tilde{T}(z^{-1}) &= 9.8521 \\ \tilde{S}(z^{-1}) &= 1 + 0.7841z^{-1} + 0.1404z^{-2} \end{aligned} \quad (35)$$

The Cruise Control problem for the leading car is following the reference trajectory with the desired speed $v_{ref}(k)$, at a discrete time k.

The Adaptive Cruise Control for the host car is following the reference trajectory with the desired distance $d_{ref}(k)$, at a discrete time k. The parameters of ACC system are based on the constant time headway policy given as $c_0=2(m)$, $t_k=0.65(s)$, and the desired distance is proportional to its velocity. For the constant spacing policy, the ACC system maintains at a fixed constant between the two cars, namely $l_0=5(m)$.

The numerical results of the car system, with proposed controller, are indicated in Figures 6 - 11. For the host car using ACC system based on the constant time headway policy, the results are depicted in Figures (6), (7) and (8) and those for the host car using ACC system based on the constant spacing policy are depicted in Figures (9), (10) and (11).

The leading car using RST algorithm follows the velocity profile introduced at reference velocity by drivers set. For the host car, Figures (6) and (9) illustrate the velocity for both constant time headway and the constant spacing policies, together with the reference velocity of the leading car. The purpose is to assess the capability of the vehicle to follow the efficiency of RST algorithm. Thus, from these figures, it can be observed that the speed profile introduced as reference for the leading car is followed, i.e., when the leading car accelerates and decelerates, the host car (ACC car) also accelerates and decelerates accordingly. However, there is a slight overshoot in the case of the constant time headway policies, as it can be observed in the zooms from Figures (6) and (9).

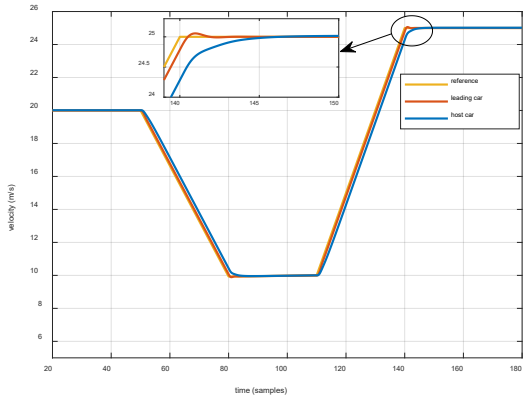

Figure 6. The velocity of the leading car and the host car based on CTHP

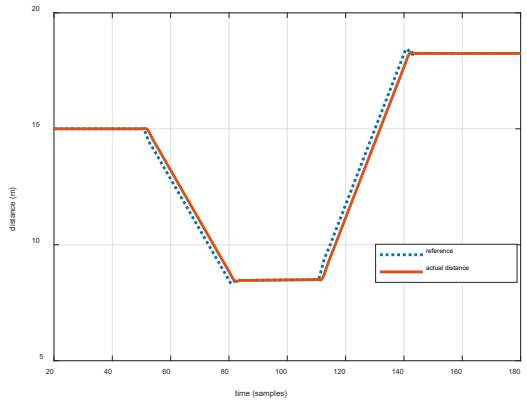

Figure 7. Inter-car distance between two cars based on CTHP

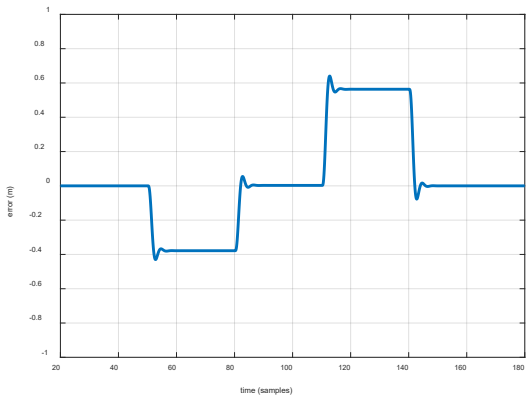

Figure 8. The distance error of the host car based on CTHP

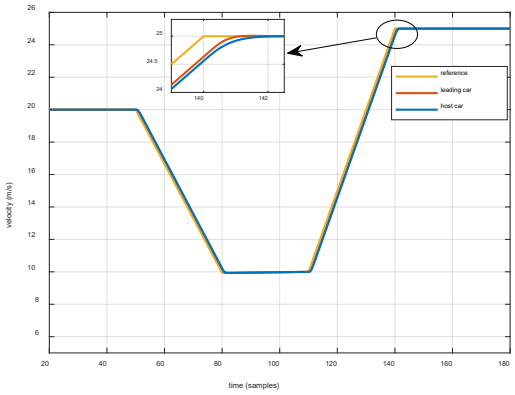

Figure 9. The velocity of the leading car and the host car based on CSP

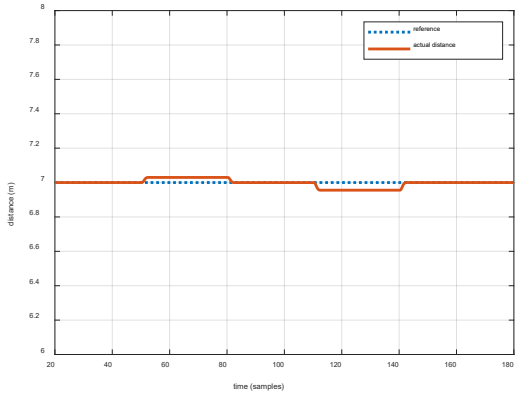

Figure 10. Inter-car distance between two cars based on CSP

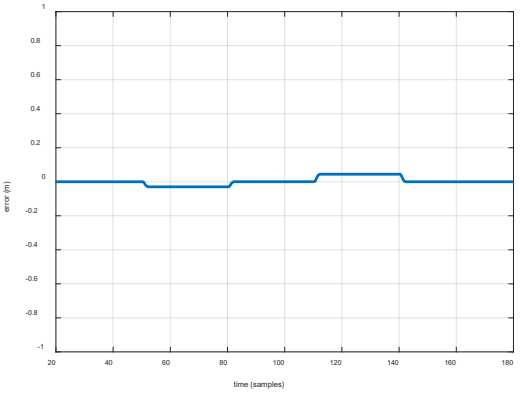

Figure 11. The distance error of the host car based on CSP

The distance error, the inter-car spacing for two cases using the constant spacing and the constant time headway policies are analyzed, and the simulation results are depicted in Figures (7), (8), (10), and (11).

The host car for these two cases maintains well the desired distance with respect to the leading car, which is expressed through distance errors (see Figures (8) and (11)). Due to the different velocities between the two cars, the spacing deviation is changing. Overall, from the resulted obtained after simulation, it can be observed that distance error of the host car for both cases converges to zero value.

For CTHP, as seen in Figure 8, the error bound of the RST controller spacing error is *-0.35* to *0.57m*. The peak value of the spacing error generated by the RST algorithm is about *0.57m*, while the peak values of the PID (Proportional Integral Derivative) algorithm is about *1.2m*. For CSP, as seen in Figure 11, the error bound of the RST controller spacing error is *-0.03* to *0.048m*. The peak value of the spacing error generated by the RST algorithm is about *0.048m*.

In the constant spacing policy case, the distance error is decreasing faster. It can be observed that the RST algorithm can reduce the peak value of the following spacing error of the car. The peak value of the spacing error reflects the performance of the control strategy under extreme conditions, such as sudden acceleration and sudden deceleration.

To sum up, the effectiveness of the proposed approach is well demonstrated, i.e., the leading car equipped with the CC system using RST algorithm has a quick response in obtaining the values closer to the reference. The host car equipped with the ACC system for two-type policies using RST algorithm has also a quick response and keeps the desired distance values.

6.2 Testing Results

The proposed method will be tested for a single lane smart car platform, in a laboratory, as it has been shown in Figure 5, in order to verify the effectiveness of its practical use.

As it has been described in section 5, the smart car platform consists of a leading car and a host car. The cars travel the distance at a very slow speed, so, by using the infrared device, the distance between the

leading car and the host car can be measured. The longitudinal speed is measured using the encoder sensor mounted on the rear wheels. The advantages of the RST structure include its simplicity and applicability in real-time systems.

This sector will focus only on the hosting car using RST controller, based on the CSP, which is a common type of strategy that shows real-time applicability. The RST algorithm will be embedded into two smart cars, where the leading car equipped with the CC system stays at the desired speed, and the host car equipped with the ACC system keeps at the desired fixed distance.

The leading car maintains speed at *0.45 m/s* during the time interval *[0, 30s]*, then decelerates from *0.45 m/s* to *0.35 cm/s* during the interval *[30s, 35s]*, and then maintains speed to *0.35m/s* during interval *[35s, 55s]*, where initial distance and desired fixed distance are set to *0.3m*.

Their velocities and the distances between the two smart cars are indicated in Figure (12) and Figure (13), respectively.From the results obtained after the simulation, as illustrated in these figures, it can be seen that the velocity tracking operates well.

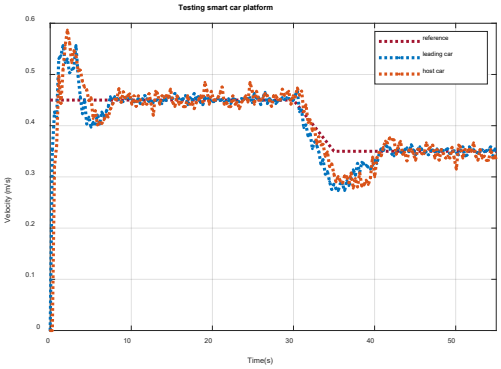

Figure 12. Velocities of the two smart cars of the host platform based on CSP

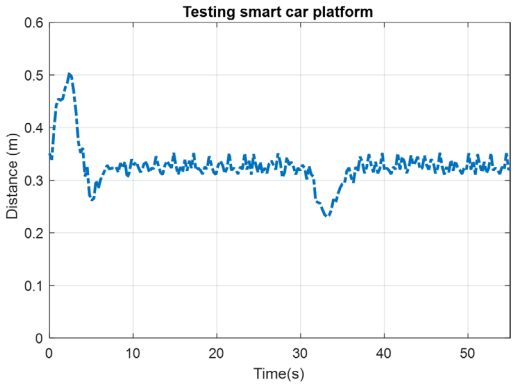

Figure 13. Inter-car distance between the two smart cars of the host platform based on CSP

Clearly, the distance related to the host smart car platform converges to the desired value, i.e., *0.32m.*

For the host smart platform, there is a little long distance at the beginning (time at *0s*) and a little overshoot after that, mainly caused by a starting voltage higher than the minimum operating voltage. From the testing results, it can be clearly seen that the RST algorithm satisfies the experiment demand for the two smart cars. However, the time delay occurred when the cars electronic control unit of the cars determined the control command for the engine driver. The devices are affected by various environmental factors, such as noise and brightness, which lead to unstable measurements. Although, it does not seriously affect the results. In general, an accurate sensor would be more appropriate, as it will improve the performance of the smart car platform.

7. Conclusion

In this study, the CC and ACC systems have been designed, simulated, and tested. Initially, the continuous time system of the mathematical models of the car was converted into discrete time system, by using the MATLAB function. Then, two discrete-time controller RST structures that apply to the CC system of the leading car and the ACC system of the host car were implemented and simulated in MATLAB. These controllers were next embedded and tested in real-time on a two-smart car platform, in a laboratory, for both CC and ACC systems, in order to evaluate the performance and effectiveness of the RST-structure for discrete-time controllers.

Based on the simulation and testing results that the system responded smoothly, the proposed method can provide solutions with good performance. The advantages of RST-structure consist in simplicity and applicability in real-time system, while, as a disadvantage, the lack of a very elaborate theoretical foundation can be observed first of all.

As a next step, in future research regarding real-time applications of this topic, a platoon of smart cars that use the hardware for vehicle-to-vehicle channels or the vehicle-to-infrastructure communication, in order to connect with other robots via WiFi, can be implemented.

Acknowledgements

Duc Lich Luu: conceptualization, methodology, writing – review & editing. **Ciprian Lupu**: formal analysis, software, hardware, validation. **Van Ga Bui**: reviewing and editing. **Nguyen Viet Hung**: software, data curation, writing – original draft. All authors have read and agreed to the published version of the manuscript.

REFERENCES

Al-Gabalawy, M., Hosny, N. S. & Aborisha, Ah. S. (2021) Model predictive control for a basic adaptive cruise control. *International Journal of Dynamics and Control*. 9(3), 1132–1143. https://doi.org/10.1007/s40435-020-00732-w.

Avdagic, Z., Cernica, E. & Omanovic, S. (2019) Adaptive neuro-fuzzy inference system based modelling of vehicle guidance. *Journal of Engineering Science and Technology*. 14(4), 2116–2131.

Ayadi, M., Haggège, J., Bouallègue, S. & Benrejeb, M. (2008) A digital flatness-based control system of a DC motor. *Studies in Informatics and Control*. 17(2), 201-214.

Chiriac, A. I. (2021) *Contributions to the implementation of control strategies in supply of high-speed rail transport*. PhD thesis, National University of Science and Technology Politehnica Bucharest.

Dawood, Y. S., Almaged, M. & Mahmood, A. (2018) Autonomous model vehicles: Signal conditioning and digital control design. *International Journal of Engineering and Innovative Technology (IJEIT)*. 8(3), 18–24. https://doi.org/10.17605/OSF.IO/SHU6K.

Dey, K. C., Yan, L., Wang, X., Wang, Y., Shen, H., Chowdhury, M., Yu, L., Qiu C. & Soundararaj, V. (2015) A review of communication, driver characteristics, and controls aspects of cooperative adaptive cruise control (CACC). *IEEE Transactions on Intelligent Transportation Systems*. 17(2), 491-509. https://doi.org/10.1109/TITS.2015.2483063.

Dokur, O. & Katkoori, S. (2023) Internet of Vehicles-Based Autonomous Vehicle Platooning. *SN Computer Science*. 5(80), 1–18. https://doi.org/10.1007/s42979-023-02391-y.

Duffney, R., Borowsky, A. & Bar-Gera, H. (2022) The effects of adaptive cruise control (ACC) headway time on young-experienced drivers' overtaking tendency in a driving simulator. *Transportation Research Part F: Traffic Psychology and Behaviour*. 86, 151-160. https://doi.org/ 10.1016/j.trf.2022.02.008.

Hu, H., Pan, W., Gao, S. & Tang, X. (2023) An Optimization-based Time-optimal Velocity Planning for Autonomous Driving. *Studies in Informatics and Control*. 32(3), 45-56. https://doi.org/10.24846/v32i3y202304.

Landau, I. D. & Zito, G. (2006) *Digital control systems: design, identification and implementation* (*Communications and Control Engineering Series*, vol. 130). New York, U.S.A., Springer.

Lupu, C., Mihai, C. C., Secuianu, F. D. & Petrescu, C. (2018) Fast Disturbance Rejection in MIMO Process Based on Algorithms Switching. In: *2018 22nd International Conference on System Theory, Control and Computing (ICSTCC)*, *10-12 October 2018, Sinaia, Romania*. New Jersey, U.S.A., Institute for Electrical and Electronics Engineers (IEEE). pp. 469-473. https://doi.org/10.1109/ICSTCC.2018.8540729.

Lupu, C., Popescu, D. & Florea, G. (2014) Supervised Solutions for Precise Ratio Control: Applicability in Continuous Production Line. *Studies in Informatics and Control*. 23(1), 53-64. https://doi.org/10.24846/v23i1y201406.

Luu, D.L., Hung, N.V. & Lupu, C. (2023) Tracking Trajectory by Using The Polynomial Method for ACC System Based on Smart Car Platform. In: *2023 17th International Conference on Engineering of Modern Electric Systems (EMES)*, *1-4 June 2023, Oradea, Romania*. New Jersey, U.S.A., Institute for Electrical and Electronics Engineers (IEEE). pp. 1-4. https://doi.org/10.1109/EMES58375.2023.10171704.

Luu, D. L., Lupu, C. & Chirita, D. (2019) Design and Development of Smart Cars Model for Autonomous Vehicles in a Platooning. In: *2019 15th International Conference on Engineering of Modern Electric Systems (EMES)*, *13-14 June 2019*, *Oradea, Romania*. New Jersey, U.S.A., Institute for Electrical and Electronics Engineers (IEEE). pp. 21-24. https://doi.org/10.1109/EMES.2019.8795199.

Ma, G., Wang, B. & Ge, S. S. (2022) Robust optimal control of connected and automated vehicle platoons through improved particle swarm optimization. *Transportation Research Part C: Emerging Technologies*. 135, 103488. https://doi.org/10.1016/j.trc.2021.103488.

Maji, P., Patra, S. K. & Mahapatra, K. (2015) Design and implementation of fuzzy approximation pi controller for automatic cruise control system. *Advances in Artificial Intelligence*. 2015(1), 1-7. https://doi.org/10.1155/2015/624638.

Pan, C., Huang, A., Chen, L., Cai, Y., Chen, L., Liang, J. & Zhou, W. (2022) A review of the development

trend of adaptive cruise control for ecological driving. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*. 236(9), 1931-1948. https://doi.org/10.1177/09544070211049.

Popescu, D., Gharbi, A., Stefanoiu, D. & Borne, P. (2017) *Process Control Design for Industrial Applications*. New Jersey, U.S.A., John Wiley & Sons.

Rajabli, N., Flammini, F., Nardone, R. & Vittorini, V. (2021) Software Verification and Validation of Safe Autonomous Cars: A Systematic Literature Review. *IEEE Access*. 9, 4797–4819. https://doi.org/ 10.1109/ACCESS.2020.3048047.

Rajamani, R. (2011) *Vehicle Dynamics and Control*. London, U.K., Springer Science & Business Media.

Sakhdari, B. & Azad, N. L. (2018) Adaptive Tube-Based Nonlinear MPC for Economic Autonomous Cruise Control of Plug-In Hybrid Electric Vehicles. *IEEE Transactions on Vehicular Technology*. 67(12), 11390–11401. https://doi.org/10.1109/TVT.2018.2872654.

Sivaji, V. & Sailaja, M. (2013) Adaptive cruise control systems for vehicle modeling using stop and go manoeuvres. *International Journal of Engineering Research and Applications*. 3(4), 2453–2456.

Stefanoiu, D., Barrio, A. M., Stoica, A. M., Constantinescu, C., Cimpoesu, E.-M. & Danciu, A. (2016) A RST design approach for the launchers flight control system. In: *Proceedings of the 6th International Conference on Astrodynamics Tools and Techniques (ICATT)*, *14-17 March 2016, Darmstadt, Germany*. pp. 32-37.

Tiganasu, A., Lazar, C., Caruntu, C. F. & Dosoftei, C. (2021) Comparative analysis of advanced cooperative adaptive cruise control algorithms for vehicular cyber physical systems. *Journal of Control Engineering and Applied Informatics*. 23(1), 82-92.

Trudgen, M., Miller, R. & Velni, J. M. (2018) Robust cooperative adaptive cruise control design and implementation for connected vehicles. *International Journal of Automation and Control*. 12(4), 469–494.

Tsai, C. C., Hsieh, S. M. & Chen, C. T. (2010) Fuzzy longitudinal controller design and experimentation for adaptive cruise control and stop&go. *Journal of Intelligent & Robotic Systems*. 59(2), 167–189. https://doi.org/10.1007/s10846-010-9393-z.

Ulsoy, A. G., Peng, H. & Çakmakci, M. (2012) *Automotive Control Systems*. Cambridge, U.K., Cambridge University Press.

Hung, N. V., N., Luu, D.L., Pham, Q.T. & Lupu, C. (2024) Comparative analysis of different spacing policies for longitudinal control in vehicle platooning. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering.* https://doi.org/10.1177/09544070241273985

Wang, Y., Wang, S., Su, C., Li, X., Zhang, Q., Zhang, Z. & Tian, M. (2024) Car-following stability improvement of cooperative adaptive cruise control based on distributed model predictive control. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*. https://doi.org/10.1177/095440702312113.

Wu, Y., Li, S.E., Cortés, J. & Poolla, K. (2019) Distributed sliding mode control for nonlinear heterogeneous platoon systems with positive definite topologies. *IEEE Transactions on Control Systems Technology*. 28(4), 1272-1283. https://doi.org/10.1109/TCST.2019.2908146.

Yang, J., Chu, D., Lu, L., Meng, Z. and Deng, K. (2024) Model-data-driven control for human-leading vehicle platoon. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*. https://doi.org/10.1177/095440702412400.

Yue, W., Guo, G. & Yuan, W.L. (2017) Bidirectional platoon control of Arduino cars with actuator saturation and time-varying delay. *Journal of Control Engineering and Applied Informatics*. 19(1), 37-48.

Zhou, F., Tao, K. & Zheng, G. (2022) Distributed cruise control and comfort evaluation for high-speed trains under moving block mode. *Journal of Physics: Conference Series*. 2203, 012062. https://doi.org/10.1088/1742-6596/2203/1/012062.