An adaptive cooperative adaptive cruise control against varying vehicle loads*

Yiming Zhang, Jia Hu*, Haoran Wang and Zhizhou Wu, *IEEE*

***Abstract*—In this paper, a universal CACC is proposed to accommodate the fact that vehicle load changes from time to time. The same vehicle could weigh differently when hauling a different number of passengers or moving a different amount of cargo. To achieve this, a dynamic matrix control-based approach is designed. The proposed controller is formulated in space domain to take advantage of the historical information of the leader to improve control accuracy and stability. It was evaluated on a Carsim-Prescan integrated simulation platform. Sensitivity analysis was conducted in terms of speed and vehicle load. Results confirm that the same proposed controller is able to maintain string stability while handling varying speeds and vehicle loads.**

I. Introduction

The cooperative and adaptive cruise control (CACC) system is developed as an enhancement of adaptive cruise control (ACC) system. It aims to minimize the minimize the speed differences among vehicles in the platoon and maintain a smaller gap between adjacent vehicles. Past studies have demonstrated that CACC has the potential to reduce traffic accident, improve comfort and increase the highway capacity[1-3].

Multiple researches have been conducted on the issues of CACC systems, among which the control problem under uncertainties has become the current hot spot[4-6]. To list a few, for control uncertainties, stochastic or adaptive MPC have been utilized to cope with control uncertainties[7]. For communication uncertainties, stochastic transmission delay and packet loss have been considered while modeling, aiming to guarantee string stability[8]. For environmental uncertainties, the platoon heterogeneity and cut-in issues are taken into account while modeling, which can improve traffic safety and alleviate oscillations of the mixed traffic flow[9].

For control uncertainties, existing research are with limitations[10, 11]. Most research model system actuation process as a first order or second order process. Since the system response of actuator is quite stochastic and nonlinear in reality, this assumption is too ideal to reality. Besides, the calibration process to handle with unknown control parameters is a little long, which further brings an obstacle for these methods to be applied to practice, since the control parameters in reality can not be obtained in advance and are varying with different loads.

For the consideration of environmental uncertainties, there are also some deficiencies among the existing researches[12]. To cope with platoon heterogeneity, many researchers have done a lot of effective work to predict the human driver's behavior. However, since human driver's driving behavior are with high stochasticity, systematic predictive error always exists and may lead to negative impact on the string stability of the platoon.

In this paper, a space-domain based approach is proposed to model the system dynamics. By utilizing the leading vehicle's historical information instead of its current information, it totally eliminates the systematic error of prediction. Besides, in order to cope with the dynamic delay of the low-level control system, a Dynamic Matrix Control (DMC) based method is proposed, which can be treated as a bridge to link low level control process and upper-level decision making process, making the controller robust to varying vehicle load. In general, the controller has the following features:

- Consider dynamic vehicle actuator delay when modeling
- Robust to varying load with self-adaptive control settings
- Ready to real-time implementation

This paper is organized as follows: Section 2 presents a brief introduction about the controller design, including the control objective, architecture, vehicle dynamics and mathematical formulation. Section 3 presents the Simulation design, Section 4 demonstrate the evaluation process and analyses its results. Section 5 provides some further discussions.

II. Controller Design

A. Overall architecture

The controller is designed to realize the vehicle platooning system. It is assumed that all vehicles in the platoon are connected and automated. Any information of a particular vehicle can be broadcasted to other vehicles, including the position, velocity, acceleration, etc. The control objective is to maintain a constant inter-vehicle time gap with guaranteed safety and driving comfort.

The diagram of the controller is designed based on a classic form, including 3 phases: Perception, Planning and Actuation. As detailed in Fig.1, the algorithm is optimized based on the

*Resrach supported by National Key R&D Program of China.

Y. Zhang is with the Key Laboratory of Road and Traffic Engineering of the Ministry of Education Tongji University, Shanghai, P. R. China, 201804 (z-emin@foxmail.com).

J. Hu is with the Key Laboratory of Road and Traffic Engineering of the Ministry of Education Tongji University, Shanghai, P. R. China, 201804 (corresponding author to provide phone: +86-13588159138; e-mail: hujia@tongji.edu.cn).

H. Wang is with the Key Laboratory of Road and Traffic Engineering of the Ministry of Education Tongji University. Shanghai, P. R. China, 201804 (709182916@qq.com).

Z. Wu is with the Key Laboratory of Road and Traffic Engineering of the Ministry of Education Tongji University, Shanghai, P. R. China, 201804 wuzhizhou@tongji.edu.cn).

Manuscript 501 submitted to 2021 32nd IEEE Intelligent Vehicles Symposium (IV).
Received April 16, 2021.

information transmitted by the leading vehicle. The process is introduced briefly as follows: firstly, the expected acceleration series would be calculated based on Gap-Regulation controller; secondly, the DMC controller would calculate a series of control increments to track the expected acceleration series; in the end, the first command would be transmitted to low level controller as the execution command.

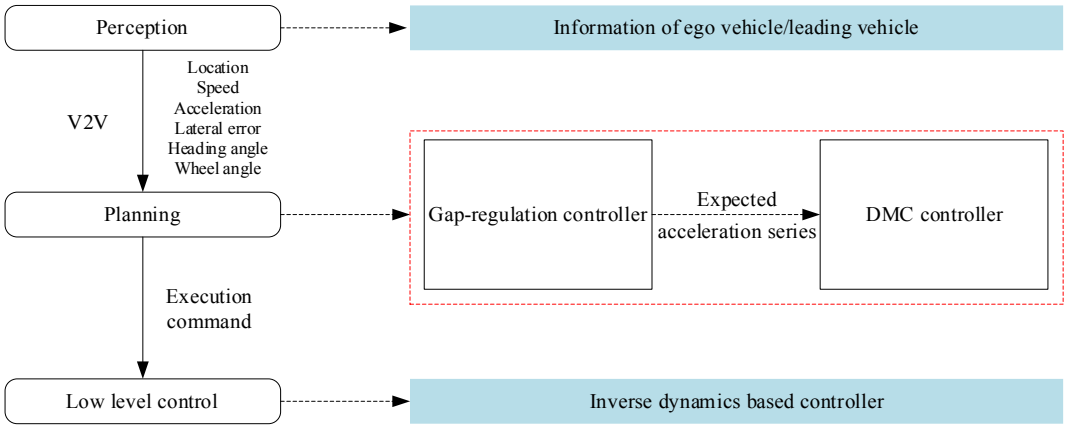


FIGURE 1 Architecture of the CACC system

B. *Gap-Regulation Controller*

1. *Longitudinal dynamics*

For longitudinal control, all vehicles in the platoon would follow the leading vehicle with a constant time gap. Mathematical vehicle dynamics model is used to be the prediction model, which is built based on space domain, where the differential of the state vector is with respect to distance instead of time. It means distance is the independent variable, and all variables, including speed and acceleration, are regarded as functions of distance, which are listed in Tab.1. Followings are the corresponding variable with respect to space domain.

Definition 1 (Slowness): Slowness is the derivative of time with respect to position, denoted by w,where

$$w = ds/dt \tag{1}$$

Definition 2 (Moderation): Moderation is the derivative of slowness with respect to position, denoted by b, where

$$b = -a/v^3 \tag{2}$$

TABLE 1

Variables	Space Domain	Time Domain
Independent variable	s (position)	t (time)
Dependent variable	t (time)	s (position)
First-order derivative	w (slowness)	v (speed)
Second-order derivative	b (moderation)	a (acceleration)

The state vector x and control vector u can be written as:

$$x^T = (g^* - g, w_f - w) \tag{3}$$

$$u = b \tag{4}$$

where g^* the expected time gap, g the actual time gap, w_f is the leading vehicle's slowness, w is the ego vehicle's slowness, b is the moderation of the ego vehicle.

The vehicle longitudinal dynamics can be given by:

$$\dot{x} = A_k x(k) + B_k u(k) + C_k \tag{5}$$

with

$$A_k = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$B_k = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

$$C_k = \begin{bmatrix} 0 \\ b_f \end{bmatrix}$$

where b_f is the moderation of leading vehicle.

Due to the cubic relation between moderation and speed, it is difficult to find a proper constraint and weight factor that is suitable for both low and high speed conditions. Beyond that, singularities are unavoidable when utilizing this system dynamics in very low speed, since infinitesimal in denominators would lead to an infinite value. To address this problem, equivalent substitution is utilized to transform the control vector form moderation to acceleration. By combining Eq.2 and Eq.5, it gives:

$$\begin{aligned} \dot{x} &= A_k' x(k) + B_k' u(k) + C_k' \\ u' &= a \end{aligned} \tag{6}$$

Manuscript 501 submitted to 2021 32nd IEEE Intelligent Vehicles Symposium (IV).
Received April 16, 2021.

where

$$A'_k = A_k$$

$$B'_k = \begin{bmatrix} 0 \\ w(k)^3 \end{bmatrix}$$

$$C'_k = \begin{bmatrix} 0 \\ -w_f(k)^3 a_f(k) \end{bmatrix}$$

among which a_f is the acceleration of leading vehicle.

To be noted, the converted system dynamics is a Linear Time-Varying model, where B'_k and C'_k would change with slowness during the prediction horizon.

2. *Mathematical formulation and solution*

The cost function of the GRC controller is modeled based on quadratic function, which is of the following form:

$$L_k = \frac{1}{2} X^T(k) Q_k x(k) + \frac{1}{2} u^T(k) R_k u(k) \quad (7)$$

With cost matrices for the longitudinal controller is as follows:

$$\mathrm{Q}_k = \begin{bmatrix} \beta_1 & 0 \\ 0 & \beta_2 \end{bmatrix} \; R_k = \beta_3 \quad (8)$$

Dynamic-programming based model predictive control (DPPC) algorithm is used to solve the algorithm. This algorithm involves the backward calculation of concomitant matrices and forward calculation of control vector and state vector. The detailed solving solution is presented in a previous study proposed by this research team[13, 14]. For reader's convenience, a brief introduction is presented as follows:

1. Calculate A_k,B'_k and C'_k for $k \in \{0,1,\dots,N\}$. Initially, the value of k=0 is set to be the default value among the prediction horizon.

2. Calculate Q_k and R_k for $k \in \{0,1,\dots,N\}$.

3. For $k = N+1$, set the concomitant matrices as follows.

$\widetilde{\mathrm{Q}}_{N+1} = Q_{N+1} \;\; \widetilde{D}_{N+1} = 0 \;\; \tilde{E}_{N+1} = 0$

4. For $k \in \{\mathrm{N}, \mathrm{N}-1, \dots, 0\}$, calculate the concomitant matrices backward, according to the following equations.

$$\widetilde{\mathrm{Q}}_k = G_k^T R_k G_k + S_k^T \tilde{Q}_{k+1} S_k + \mathrm{Q}_k \quad (9)$$

$$\widetilde{\mathrm{D}}_k = G_k^T R_k H_k + S_k^T \tilde{Q}_{k+1} T_k + S_k^T \overline{D}_{k+1} \quad (10)$$

$$\widetilde{\mathrm{E}}_k = \frac{1}{2} H_k^T R_k H_k + \frac{1}{2} T_k^T \tilde{Q}_{k+1} \mathrm{T}_k \quad (11)$$

$$\widetilde{\mathrm{Q}}_k = \mathrm{G}_k^T \mathrm{R}_k \mathrm{G}_k + \mathrm{S}_k^T \widetilde{\mathrm{Q}}_{k+1} \mathrm{S}_k + \mathrm{Q}_k \quad (12)$$

$$\widetilde{\mathrm{D}}_k = \mathrm{G}_k^T \mathrm{R}_k \mathrm{H}_k + \mathrm{S}_k^T \widetilde{\mathrm{Q}}_{k+1} \mathrm{T}_k + \mathrm{S}_k^T \widetilde{\mathrm{D}}_{k+1} \quad (13)$$

$$\widetilde{\mathrm{E}}_k = \frac{1}{2} \mathrm{H}_k^T \mathrm{R}_k \mathrm{H}_k + \frac{1}{2} \mathrm{T}_k^T \widetilde{\mathrm{Q}}_{k+1} \mathrm{T}_k \quad (14)$$

With

$$\mathrm{P}_k = (\mathrm{R}_k + \mathrm{B'}_k^T \widetilde{\mathrm{Q}}_{k+1} \mathrm{B'}_k)^{-1} \quad (15)$$

$$\mathrm{G}_k = -\mathrm{P}_k \mathrm{B}_k^T \widetilde{\mathrm{Q}}_{k+1} \mathrm{A}_k \quad (16)$$

$$\mathrm{H}_k = -\mathrm{P}_k \mathrm{B'}_k^T (\widetilde{\mathrm{Q}}_{k+1} \mathrm{C'}_k + \widetilde{\mathrm{D}}_{k+1}) \quad (17)$$

$$\mathrm{S}_k = \mathrm{A}_k + \mathrm{B'}_k \mathrm{G}_k \quad (18)$$

$$\mathrm{T}_k = \mathrm{B'}_k \mathrm{H}_k + \mathrm{C'}_k \quad (19)$$

5. For $k \in \{0,1,\dots,N\}$, calculate control vector and state vector forward, respectively according to the following equations.

$$u(k) = G_k x(k) + H_k \quad (20)$$

$$\mathrm{x(k+1)} = \mathrm{S}_k x(k) + T_k \quad (21)$$

6. Determine whether the results converge, if not, re-execute the previous steps.

The matrices, A_k, B'_k and C'_k, which is required in step 1, have already been discussed in the modelling of longitudinal dynamics.

C. DMC controller

DMC control method is applied to take consideration of the actuator delay of the low-level control system. The key point of the DMC controller is to find the optimal control increments, in order to track the desired trajectory of the control vector as close as possible. In order to cope with stochasticity, an on-line self-adaptive regulator is introduced while modeling. Followings are the introduction of DMC controller:

The prediction model is built based on step response of low-level controller, which can be describe as:

$$\mathrm{A}_d = [a_1, a_2, a_3 \dots a_N]$$

where a_i represents the value of step response at different sampling point (as is shown in Fig.2), N is the prediction horizon of the model.

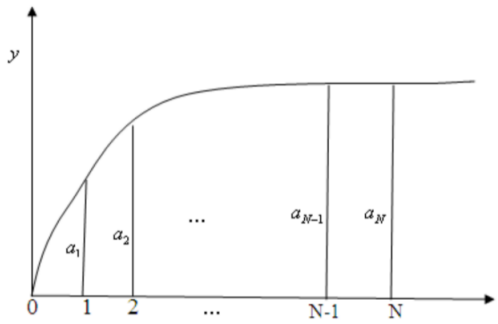


FIGURE 2 Step response of different points

The system dynamics can be written as:

$$\hat{y}(k+1) = \hat{y}_0(k) + a_i u(k) \quad (22)$$

Similarly, if there are M control increments applied to the control member, then the output of the system can be forecasted as:

$$\hat{y}_i(k+i) = \hat{y}_0(k+i) + \sum_{j=1}^{M} a_{i-j+1} \Delta u(k+j-1) \quad (23)$$

where Δu is the control increment, M is the number of steps in control domain, obviously $M < N$.

Considering the objective of the DMC controller is to track the desired trajectory of control vector as precise as possible, the performance index can be modeled as:

$$(\mathrm{k}) = \sum_{i=1}^{N} q_i [y_r(k+i) - \hat{y}_M(k+i)]^2 + \sum_{j=1}^{M} r_j \Delta u^2(k + j - 1) \quad (24)$$

J where q_i and r_j are weight factors, y_r is the control vector given by Gap Regulation Controller.

Based on Pontryagin's minimum principle, the optimal control increments series can be given as:

$$\Delta u_M(k) = (A^T Q A + R)^{-1} A^T Q [y_{rp}(k) - \hat{y}_{r0}(k)] \quad (25)$$

Considering the system dynamic in real world is time-varying and always exists disturbance, the prediction value may not be in accordance with the value in real world. Thus, an on-line self-adaptive regulator is introduced to calibrate system errors.

Defining $e(k+1)$ as the error between prediction value and actual output at step $k+1$, it can be expressed as:

$$e(k+1) = y(k+1) - \hat{y}_M(k+1) \quad (26)$$

where $y(k+1)$ is the actual output at step $k+1$, $\hat{y}_M(k+1)$ is the predicted output at step $k+1$.

The calibrated output can be given as:

$$\hat{y}_{calib} = \hat{y}_M(k+1) + he(k+1) \quad (27)$$

where h is calibration vector, it can be expressed as:

$$h = [h_1, h_2, \dots, h_N]^T \quad (28)$$

D. Low-level controller

Inverse-dynamics based method is utilized to construct low-level controller, where the command from upper-level controller would be translated into throttle and brake pressure. The condition of acceleration and deceleration are modeled respectively, of which the diagram is shown in Fig.3.

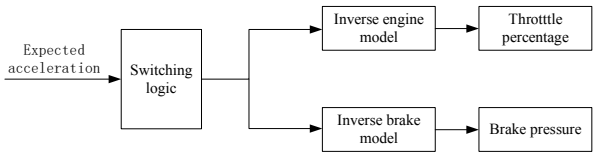


FIGURE 3 Step response of different points

1. *Acceleration control mode*

When switching to acceleration control mode, the controller would perform the following procedures:

STEP 1: Calculate the expected engine torque according to expected acceleration

STEP 2: Translate expected engine torque to throttle percentage based on engine MAP

For convenience, the following assumptions are made:

Assumption 1: Vehicle is running in flat road, slope angle is not within consideration.

Assumption 2: Rolling resistance coefficient and air density always stay constant.

Assumption 3: The wind speed is negligible.

Assumption 4: The deformations of transmission system and tyres are negligible.

Based on force analysis, the longitudinal dynamic equation can be given as:

$$ma_{exp} = F_t - F_f - F_w \quad (29)$$

where F_t is the driving force generated by engine, F_f is the rolling resistance, F_w is the air resistance.

The rolling resistance F_f can be expressed as:

$$F_f = mgf \quad (30)$$

where m is total weight, g is gravitational acceleration, f is rolling resistance coefficient.

The air resistance can be expressed as:

$$F_w = \frac{1}{2} C_D A \rho_w v^2 \quad (31)$$

where C_D is the air resistance coefficient, A is the windward area, ρ_w is air density, v is longitudinal speed.

Based on the aforementioned assumptions, the driving force can be given as:

$$F_t = \frac{\eta_T T_{tq} \iota(\frac{w_t}{w_e}) i_g i_0}{r} \quad (32)$$

where η_T is the mechanical efficiency, T_{tq} is the engine torque, w_t is the turbine speed, w_e is engine speed, i_g is gear ratio of transmission, i_0 is gear ratio of main speed reducer, r is rolling radius, ι is eigenfunction of torque converter.

Defining

$$K_d = \frac{\eta_T T_{tq} \iota(\frac{w_t}{w_e}) i_g i_0}{r} \quad (33)$$

Combining Eq.29-33, the expected torque can be given as:

$$T_{tq} = \frac{ma_{exp} + F_f + F_w}{K_d} \quad (34)$$

Based on vehicle engine MAP, the corresponding throttle percentage can be given as:

$$\alpha = f(T_{tq}, w_e) \quad (35)$$

2. *Deceleration control mode*

When switching to deceleration control mode, the controller would perform the following procedures:

Manuscript 501 submitted to 2021 32nd IEEE Intelligent Vehicles Symposium (IV).
Received April 16, 2021.

STEP 1: Calculate the expected braking torque according to expected acceleration.

STEP 2: Translate expected braking torque to braking pressure based on inverse brake model.

Based on force analysis, the longitudinal dynamic equation can be given as:

$$ma_{exp} = -(F_b + F_f + F_w) \quad (36)$$

where F_f is braking force.

When braking force is less than the maximum braking force provided by road surface, the following equation holds:

$$F_b = \frac{T_{bf} + T_{br}}{r} \approx K_b P_{exp} \quad (37)$$

where T_{bf} is the sum of braking force applied to front wheels, T_{br} is the sum of braking force applied to rear wheels, K_b is a conversion coefficient, P_{exp} is expected braking pressure.

Thus, the expected braking pressure can be given as:

$$P_{exp} = F_b / K_b \quad (38)$$

3. *Switching logic of accelerate/decelerate*

To ensure driving comfort and fuel economy, the frequent switching between throttle and brake should be avoided. To achieve this, a natural deceleration curve is established, which describes the relationship between speed and deceleration when vehicle is running without throttle/brake input. Transition region is set across the curve with a $2\,\Delta h$ depth to avoid frequent switching, as is shown in Fig.4, where $\Delta h = 0.2\ s$. The switching logic is:

i) Accelerate when $a_{exp} \geq a + \Delta h$

ii) Decelerate when $a_{exp} \leq a - \Delta h$

iii) No control input when $a + 2\,\Delta h \geq a_{exp} \geq a - 2\,\Delta h$

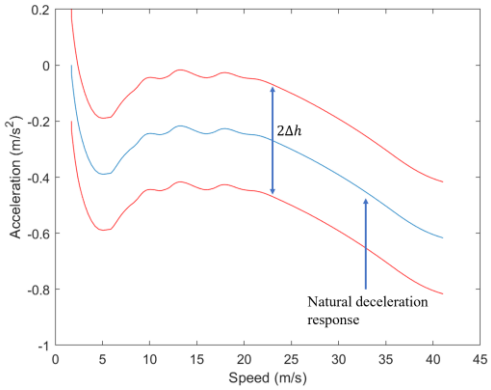


FIGURE 4 Switching logic of accelerate/decelerate

III. Simulation Design

In order to validate the performance and reliability of the proposed algorithm, a software-in-the-loop simulation platform is developed based on Prescan/Carsim. As is shown in the figure below, Prescan provides traffic environment, Carsim provides vehicle kinetic and dynamic model.

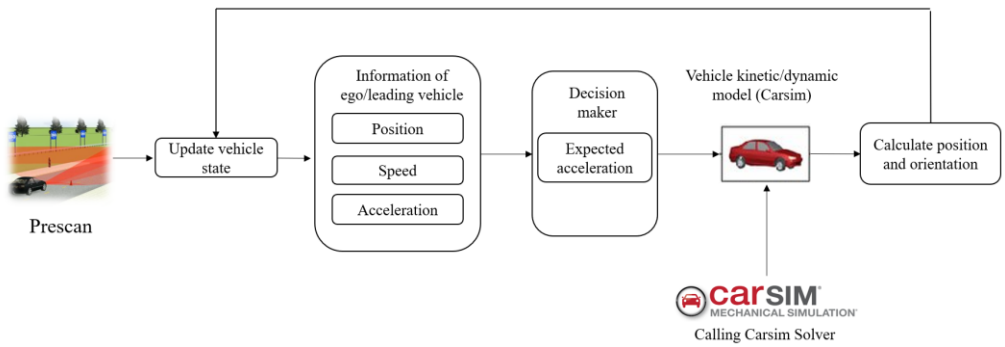


FIGURE 5 Architecture of simulation platform

In simulation environment, Mazda z6 is chosen to be the experimental vehicle running in Prescan and Carsim. Each vehicle is equipped with V2X facilities (support both transmitting and receiving) and on-board computing unit. The vehicle platoon consists of 3 vehicles. The speed profile of the leading vehicle is pre-defined. The traffic environment in simulation is long straight road. Followings are the experimental settings for simulation:

- Desired time gap:1 $second$
- Optimization horizon: the corresponding distance of next 10 $second$
- The weighting factors for Gap-Regulation controller:$[200,1,1]^T$
- The weighting factors for DMC controller: $[10,10,10,10,10,\overbrace{1,1,\dots,1}^{n=45}]^T$
- Acceleration range is [-5,3] m/s^2

In order to standardize the driving cycle, the maneuvering process has been divided into 4 stages, including i) **Acceleration**, where vehicle will accelerate with constant acceleration; ii) **Oscillation**, where the vehicle will accelerate and decelerate repeatedly with a constant acceleration/deceleration; iii) **Uniform motion**, where the vehicle will drive with a constant speed; iv) **Deceleration**, where the vehicle will decelerate with a constant deceleration. In simulation environment, the leading vehicle is assumed to

Manuscript 501 submitted to 2021 32nd IEEE Intelligent Vehicles Symposium (IV).
Received April 16, 2021.

be ideal, indicating that there would be no jerk limitation of the leading vehicle.

In order to test the performance and robustness of the proposed algorithm, different parameters, including running speed and vehicle loads, are selected to conduct sensitivity analysis. To be specific, running speed includes 40-120 *km/h* (40 increments), vehicle loads includes 0- *500 kg* (250 increments).

IV. Evaluation

A. Measurement of Effectiveness (MOE)

In order to evaluate the performance of the algorithm intuitively, the difference between expected time gap and actual time gap is chosen as the key index. To be detailed, there are two variables

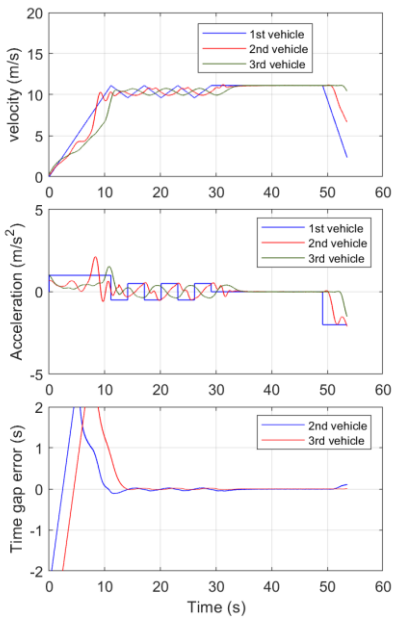
Gap-Error.1-2: The time gap error of vehicle 1 and 2. The unit is in seconds
Gap-Error.1-3: The time gap error of vehicle 1 and 3. The unit is in seconds.

B. Evaluation results

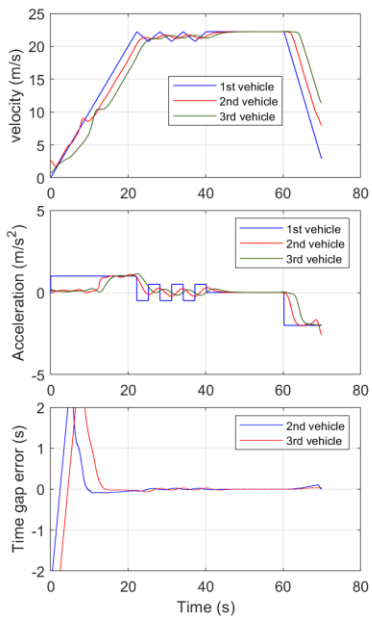
Generally, the evaluation results confirm the validity of the proposed vehicle platooning algorithm. The results show that the platoon is able to keep a constant time gap with high accuracy. The oscillations produced by the human-driven vehicle can be smoothed along the platoon.

1. Straight driving with different speed

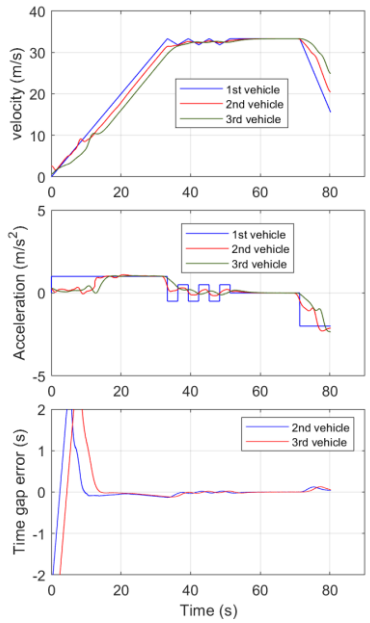
The results of straight driving with different speed are detailed in Fig.6. The maximum time gap error is 0.03 s. Results show that time gap error is not sensitive to driving speed. It demonstrates that the platoon is able to keep a constant time gap under various speed conditions. To be noted, the speed oscillations of vehicle 2 and vehicle 3 is less than vehicle 1, it confirms the string stability of the proposed controller. It is worth noting that Gap-Error.1-3 is less than Gap-Error.1-2, the reason for that mainly attribute to the greater expected time gap with the leading vehicle, giving more history information to make regulations.



Performance under the condition of 40 *km/h* and 0.5 m/s^2 acceleration



Performance under the condition of 80 *km/h* and 0.5 m/s^2 acceleration



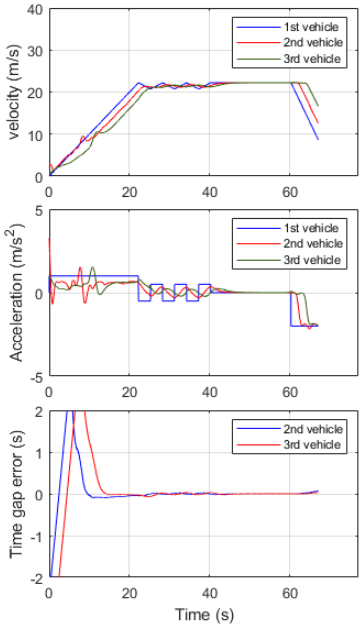
Performance under the condition of 120 *km/h* and 0.5 m/s^2 acceleration

FIGURE 6 Performance Trajectories during Straight Cruising under different speed conditions

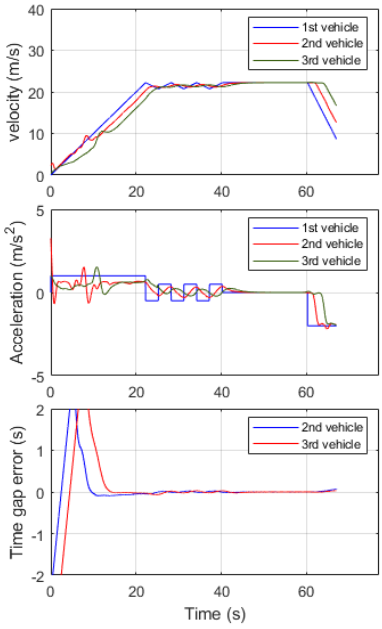
2. Straight driving with different loads

The results of straight driving with different loads are detailed in Fig.7. Based on the results, the performances under different loads are approximately the same, of which the maximum time gap is 0.04 s. It demonstrates that the platoon is able to keep a constant time gap with high speed under various vehicle loads. Besides, it is worth noting that the oscillation acceleration of vehicle 2 and vehicle 3 is always less than vehicle 1, which further confirms the string stability of the platoon.

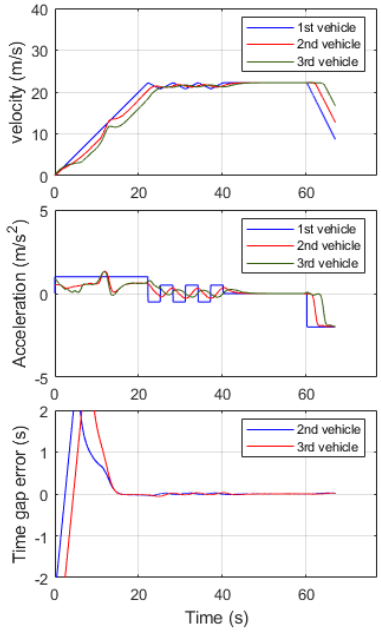
Manuscript 501 submitted to 2021 32nd IEEE Intelligent Vehicles Symposium (IV).
Received April 16, 2021.



Performance under the condition of 80 *km/h* and 0kg load



Performance under the condition of 80 *km/h* and 250 kg load



Performance under the condition of 80 *km/h* and 500 kg load

V. Conclusion

In this paper, a dynamic matrix control based approach is utilized to deal with the non-linearity and uncertainty of vehicle dynamics. The proposed controller is modeled in space domain to use the historical information of the predecessor. Simulation results show that the controller can guarantee high control accuracy and string stability under various conditions. Detailed investigation on the results reveals that:

- Under the control of the proposed CACC controller, maximum time gap error is within 0.1 s under various conditions.
- Time gap error is not sensitive to the variation of speed, but sensitive to oscillation acceleration.
- String stability is confirmed. It indicates that the proposed controller is able to smooth the oscillation produced by human drivers.

The controller proposed in this paper does not take into account the lateral control, which limit the potentiality to lessen the drivers' driving load. Further work could consider coupling longitudinal control and lateral control together. This enhancement would further improve the practicability of the CACC system.

References

1. Ploeg, J., et al. *Design and experimental evaluation of cooperative adaptive cruise control*. in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. 2011. IEEE.
2. Shladover, S.E., et al., *Cooperative adaptive cruise control: Definitions and operating concepts.* Transportation Research Record, 2015. **2489**(1): p. 145-152.
3. Shladover, S.E., D. Su, and X.-Y. Lu, *Impacts of cooperative adaptive cruise control on freeway traffic flow.* Transportation Research Record, 2012. **2324**(1): p. 63-70.
4. Shao, Y. and Z. Sun. *Robust eco-cooperative adaptive cruise control with gear shifting*. in *2017 American Control Conference (ACC)*. 2017. IEEE.
5. Trudgen, M., R. Miller, and J.M. Velni, *Robust cooperative adaptive cruise control design and implementation for connected vehicles.* International Journal of Automation and Control, 2018. **12**(4): p. 469-494.
6. Zhao, S. and K. Zhang, *A distributionally robust stochastic optimization-based model predictive control with distributionally robust chance constraints for cooperative adaptive cruise control under uncertain traffic conditions.* Transportation Research Part B: Methodological, 2020. **138**: p. 144-178.
7. Zhou, Y., et al., *Rolling horizon stochastic optimal control strategy for ACC and CACC under uncertainty.* Transportation Research Part C: Emerging Technologies, 2017. **83**: p. 61-76.
8. Zhang, Y., et al., *Control Design, Stability Analysis, and Traffic Flow Implications for Cooperative Adaptive Cruise Control Systems with Compensation of Communication Delay.* Transportation Research Record, 2020. **2674**(8): p. 638-652.
9. Wang, C. and H. Nijmeijer. *String stable heterogeneous vehicle platoon using cooperative adaptive cruise control*. in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. 2015. IEEE.
10. Zhu, Y., D. Zhao, and Z. Zhong, *Adaptive optimal control of heterogeneous CACC system with uncertain dynamics.* IEEE Transactions on Control Systems Technology, 2018. **27**(4): p. 1772-1779.
11. Moser, D., et al. *Cooperative adaptive cruise control applying stochastic linear model predictive control strategies*. in *2015 European Control Conference (ECC)*. 2015. IEEE.
12. Gong, S. and L. Du, *Cooperative platoon control for a mixed traffic flow including human drive vehicles and connected and autonomous vehicles.* Transportation research part B: methodological, 2018. **116**: p. 25-61.
13. Y. Bai, Y.Z., and J. Hu, *Cooperative Lane Changing Strategies for Connected Automated Vehicles under Model Predictive Control Framework*, in *14th International Symposium on Advanced Vehicle Control(AVEC)*. 2018: Bei Jing，China.

Manuscript 501 submitted to 2021 32nd IEEE Intelligent Vehicles Symposium (IV).
Received April 16, 2021.

14. Yu Zhang, Y.B., Meng Wang, Jia Hu*, *Chang-Hu's Optimal Motion Planning Framework for Cooperative Automation: Mathematical Formulation, Solution, and Applications.* Transportation Research Board Annual Meeting, 2020.