

Model-based Predictive Control implementation for Cooperative Adaptive Cruise Control

António Lopes¹, Rui Esteves Araújo²

INESC TEC – INESC Technology and Science, Faculty of Engineering, University of Porto, Porto, Portugal ¹(antonio.lobes@fe.up.pt); ²(raraujo@fe.up.pt)

Abstract

The automation of road vehicles has become a necessity to improve the efficiency and safety of this system. In a vehicle formation it is important to maintain a safety distance between the vehicles. The control of a vehicle spacing distance and longitudinal velocity can be achieved through the implementation of a model-based predictive controller. This implementation of a cooperative adaptive cruise control allows the access of another vehicle state information through vehicular communication technology and promote state prediction and ultimately system stability. The optimization algorithm performs the computation of the control input in a control horizon window and ensures that the spacing error takes only positive values. The results of the proposed controller are evaluated through the computational tool Simulink in the two-vehicle platoon. The controller is implemented in the precedent vehicle. To assess the performance of the proposed controller different control parameters and constraints were used.

Subject Headings. Automation, Computer System, Information System

Author Keywords. Cruise Control, Intelligent vehicle, Vehicle-to-vehicle communication

1. Introduction

In the last few decades, the amount of road traffic as increased dramatically, which led to a clear road congestion issue that has become more relevant in recent years (Dey et al. 2015; van Arem, van Driel, and Visser 2006). Another negative aspect that arises from the increasing traffic demand is the energy consumption and the air pollution which is one of the most relevant problems in the 21st century society (van Arem, van Driel, and Visser 2006).

Besides the environmental issues, there are also safety issues involved with this particular problem, the risk of accidents and passenger's safety is also a great concern. It is well known that many car accidents that occur are due to human error and poor driver response (Vahidi and Eskandarian 2003; Dey et al. 2015). Therefore, it is clear that a robust automated system that provides a more reliable control input is needed and that this can provide a safer and more efficient solution (Vahidi and Eskandarian 2003; a R. Girard et al. 2001).

At the turn of the century, the issue of automated or intelligent transportation has become more relevant and there has an effort to improve highway capacity and safety (Vahidi and Eskandarian 2003). During this time a research and development program of the University of California has founded, the California Partners for Advanced Transportation Technology (PATH), to deal with intelligent transportation systems. This multi-disciplinary group is one of the most mature teams dealing with this issue and has published various works in this particular field. Among their research it is possible to identify relevant work from complex Automated Highway systems and string stability, to cruise control systems (Godbole and

Lygeros 1994; D. Swaroop and Hedrick 1996; Darbha Swaroop, Karl Hedrick, and Choi 2001; a R. Girard et al. 2001).

The cruise control system has been a well-known control already installed in many vehicles that allow a constant speed control in a vehicle system. The adaptive or intelligent cruise control is a more complex system that enables the car to follow the vehicle ahead at a constant safety distance (Shakouri and Ordys 2014; A. R. Girard, Spry, and Hedrick 2005). The intelligent cruise control has studied by authors like Lygeros who proposed a linear control design for the system in the 90's decade (Godbole and Lygeros 1994). Other authors like Swaroop were among the first to pursuit an automated highway system, studying the string stability of a vehicle formation (D. Swaroop and Hedrick 1996). Some authors like Payman have studied gain scheduling Proportional-Integral (PI) and Linear Quadratic (LQ) controller to cope with this problem (Payman 2011). In more recent years, nonlinear Model Predictive control (MPC) design have also been proposed for adaptive cruise control by authors like Payman and Lou (Shakouri and Ordys 2014; Luo et al. 2010).

The recent advance in vehicle-to-vehicle communication has once again renew the discussion of adaptive cruise control and introducing a new paradigm, the cooperative adaptive cruise control (CACC) (A. R. Girard, Spry, and Hedrick 2005). In this new approach, the vehicle control is not the only sensor based, but has also accesses of other vehicle information through the vehicular network, allowing sensor fusion algorithms (Vinel, Lan, and Lyamin 2015; A. R. Girard, Spry, and Hedrick 2005). In other words, CACC makes the vehicle follow its predecessor at a close but safe distance and uses also information received from other vehicles to realize its operation. In figure 1 a schematic representation of the working principle of CACC is shown.

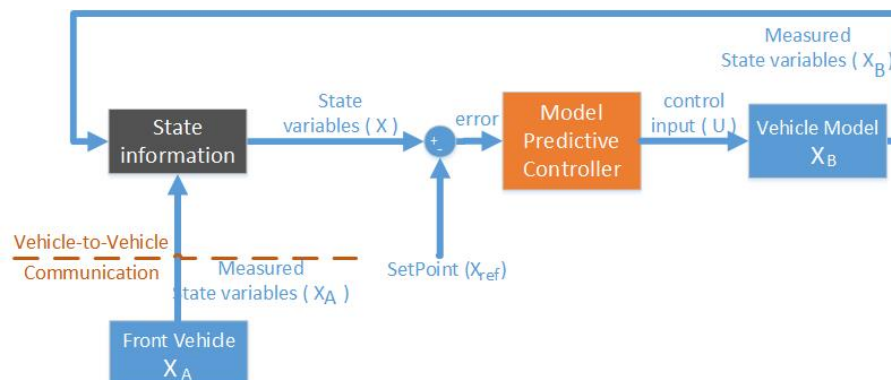


Figure 1: Control Structure for Cooperative Adaptive Cruise Control

In this paper, it presents a model predictive control approach for a CACC. The document is structured as follows: in Section 2 the problem is completely defined, section 3 presents the proposed model predictive control design, the results are presented in Section 4 as well its discussion and finally the conclusions and future work will be presented in Section 5.

2. Problem Formulation

In the vehicle system the controller for the adaptive cruise control is defined by two layers of control (Luo et al. 2010). The lower layer, which provides the throttle and brake command to track the vehicle acceleration and an upper layer of control that define the vehicle dynamic in order to follow the vehicle in front (Luo et al. 2010).

According to (Godbole and Lygeros 1994) it is possible to model the dynamic of the vehicle position x as a third order ordinary differential equation defined by (Godbole and Lygeros 1994):

$$\ddot{x} = b(\dot{x}, \ddot{x}) + a(\dot{x})\zeta \quad (1)$$

Where $a(\dot{x})$ and $b(\dot{x}, \ddot{x})$ represent nonlinear functions of the state variables of the vehicle, with $(\dot{x}) \neq 0$, and ζ as the control variable of the lower layer. For the present analysis, it will be neglected the dynamic of the nonlinear functions b and a and will be assumed a feedback linearization control law defined by (Godbole and Lygeros 1994):

$$\zeta = \frac{-b(\dot{x}, \ddot{x}) + u}{a(\dot{x})} \quad (2)$$

As a result, the vehicle jerk is defined as the input u as presented in (3).

$$\ddot{x} = u \quad (3)$$

The controller for the resulting linear dynamic will be constraint by the engine, tire and road conditions:

$$\begin{aligned} \dot{x} &\in [v_{\min}, v_{\max}], \\ \ddot{x} &\in [a_{\min}, a_{\max}], \\ u &\in [u_{\min}, u_{\max}], \end{aligned} \quad (4)$$

In this operation conditions it is assumed that the vehicle will not go backward, therefore $v_{\min} > 0$, and the maximum speed of the vehicle v_{\max} is imposed by the engine torque limitation.

The upper layer controller will be defined to closely track the requested reference, ensuring the same longitudinal velocity of the vehicle ahead and assuring at the same time a safe relative distance between the two vehicles.

In this system formulation, it is considered that there is a cooperative coordination between the two cars and that it is possible for the vehicle to track the state variables of the other vehicles through vehicle-to-vehicle communication (Vinel, Lan, and Lyamin 2015). The real time application issues associated with vehicular communication technology are defined in (Dey et al. 2015; Vinel, Lan, and Lyamin 2015) and will not be considered in the presented problem. A model predictive control (MPC) synthesis is adopted to design the CACC. The control law formulation will be discussed in the next section.

3. Design of Model Predictive Control

In a MPC, the design objective is to compute a trajectory of a future variable to optimize the future behavior of the plant output (Wang 2009). In this particular problem, we wish to compute the system jerk input in order to control the vehicle longitudinal position, velocity and acceleration. This optimization is held in a finite prediction horizon window by estimating the behavior of the system in a determined time horizon. The design of the controller is based on the mathematical model of the plant of the system.

The MPC method is set by two structuring algorithms (Wang 2009; Luo et al. 2010): the state prediction by means of the state-space model of the system and the control variable computation through an optimization algorithm.

3.1. System model

The MPC design requires a model of the relevant dynamics to use as a prediction model. Figure 2 is presented the system composed by two vehicles and the state variables associated with the system. A linear model to represent the vehicle dynamics of each vehicle within the platoon with relative distance and velocity of two vehicles is presented.

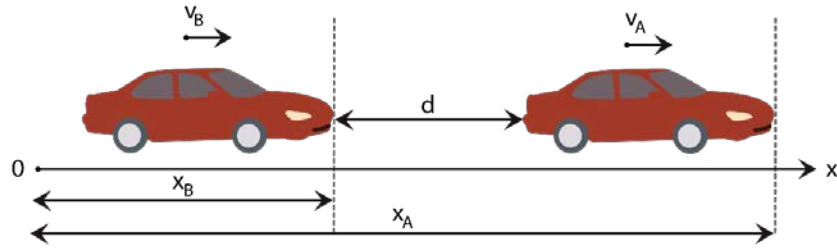


Figure 2: System of two vehicles

As it can be seen from the figure 2, vehicle A is the vehicle in front and B the preceding vehicle. The platoon is assumed to move in a straight line. From an inertial coordination frame, the vehicles A and B are at a distance x_A and x_B from zero and follow at a velocity v_A and v_B respectively.

Three state variables have been used in the platoon model. They are, the relative distance $d(n)$, the relative velocity $v(n)$ and the relative acceleration $a(n)$ between the two vehicles.

The discrete model is obtained by using the forward Euler approximation with truncated Taylor expansion. Then, we can explicitly derive the discrete-time equations of the relative distance, the relative velocity and the relative acceleration as (Luo et al. 2010):

$$\begin{aligned} d(n+1) &= d(n) + v(n)T_s - \frac{1}{2}a(n)T_s^2 \\ v(n+1) &= v(n) - a(n)T_s \\ a(n+1) &= a(n) + u(n)T_s \end{aligned} \quad (5)$$

where $x(n)$ define the state value in the n th step time ($x(n) \equiv x(t - t_n)$) given by $T_s = t_{n+1} - t_n$

The discrete-time state space model is presented in (6)

$$\begin{aligned} x(n+1) &= Ax(n) + B\Delta u(n) \\ y(n+1) &= Cs(n) \end{aligned} \quad (6)$$

with

$$A = \begin{bmatrix} 1 & T_s & -0.5T_s^2 \\ 0 & 1 & -T_s \\ 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ T_s \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (7)$$

The system output ($y(n)$) will be exactly the state vector as it is assumed that the vehicle has access to the complete state information for the purpose of feedback control.

3.2. State Prediction

The state space matrix (A,B,C) defined in (7) was used to compute the future state variables of the system(Wang 2009):

$$\begin{aligned} x(n+1|n) &= Ax(n) + B\Delta u(n) \\ x(n+2|n) &= Ax(n+1) + B\Delta u(n+1) \\ &= A^2x(n) + AB\Delta u(n) + B\Delta u(n+1) \\ &\vdots \\ x(n+N_p|n) &= A^{N_p}x(n) + A^{N_p-1}B\Delta u(n) + A^{N_p-2}B\Delta u(n+1) \\ &\quad + \dots + A^{N_p-N_c}B\Delta u(n+N_c-1) \end{aligned} \quad (8)$$

The system output variables can also be computed as presented in (Wang 2009):

$$\begin{aligned} y(n+1|n) &= CAx(n) + CB\Delta u(n) \\ y(n+2|n) &= CA^2x(n) + CAB\Delta u(n) + CB\Delta u(n+1) \\ &\vdots \\ y(n+N_p|n) &= CA^{N_p}x(n) + CA^{N_p-1}B\Delta u(n) + CA^{N_p-2}B\Delta u(n+1) \\ &\quad + \dots + CA^{N_p-N_c}B\Delta u(n+N_c-1) \end{aligned} \quad (9)$$

Where N_P and N_C define the prediction and control horizon window respectively(Wang 2009). From (8) and (9) it becomes clear that the prediction of the future value of the state variables ($y(n + j)$, with $j = [0,1, \dots, N_C - 1]$) is performed at the expense of the current state information, the state-space model and the future control input vector given by (10)(Wang 2009).

$$\Delta U = [\Delta u(n), \Delta u(n + 1), \Delta u(n + 2) \dots \Delta u(n + N_C - 1)] \quad (10)$$

The estimation of the system output is expressed as (11) (Wang 2009):

$$Y = Fx(n) + \Phi \Delta U \quad (11)$$

Where,

$$Y = \begin{bmatrix} y(n + 1|n) \\ y(n + 2|n) \\ \vdots \\ y(n + N_P|n) \end{bmatrix}, \quad (12)$$

$$F = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{N_P} \end{bmatrix}, \quad \Phi = \begin{bmatrix} CB & 0 & 0 & \dots & 0 \\ CAB & CB & 0 & \dots & 0 \\ CA^2B & CAB & CB & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{N_P-1}B & CA^{N_P-2}B & CA^{N_P-3}B & \dots & CA^{N_P-N_C}B \end{bmatrix}$$

3.3. Control Computation

The computation of the control variable is fundamental to the implementation of the predictive control system. The objective is to calculate the predicted output in such a way that the state variables and output variables converge to a pre-determined set-point(Wang 2009). This objective can be translated to find the most appropriated control vector ΔU by defining a cost function that reflects the control objective as (Wang 2009):

$$J = (R_s - Y)^T(R_s - Y) + \Delta U^T G \Delta U \quad (13)$$

Where R_s is the set-point vector that contain the information of the desired state for the N_P prediction horizon(Wang 2009). The first term of the cost function J (13) expresses the minimization of the error between the predictive output and the reference R_s (Wang 2009). The second term restrict the value of ΔU through the diagonal gain matrix $G = gI_{N_C \times N_C}$, $\forall g \geq 0$. This gain can restrict the value of the output control variable and may be used as a soft constraint (Wang 2009).

By replacing the result in (11) in the cost function in (13), J can be rewritten as (Wang 2009):

$$J = (R_s - Fx(n))^T(R_s - Fx(n)) + 2\Delta U^T \Phi^T(R_s - Fx(n)) + \Delta U^T(\Phi^T \Phi + G)\Delta U \quad (14)$$

The derivative of the cost function is defined by (Wang 2009):

$$\frac{dJ}{d\Delta U} = -2\Phi^T(R_s - Fx(n)) + 2(\Phi^T \Phi + G)\Delta U \quad (15)$$

The minimum of the cost function J is obtained if

$$\frac{dJ}{d\Delta U} = 0 \quad (16)$$

From (16) it is clear that the solution for the control signal is defined by (Wang 2009):

$$\Delta U = (\Phi^T \Phi + G)^{-1} \Phi^T(R_s - Fx(n)) \quad (17)$$

Although the computed solution is correct, in fact, there are no hard constraints associated with the control variable, state variables or output of the system, which may lead to an

impractical solution for the system. To achieve a valid solution the optimization algorithm must accommodate the system constraints as well as the physical limits of the system.

In order to obtain a solution that restrict the system to its physical limits and, at the same time, maintain the system stability, the problem can be proposed as a quadratic programming optimization problem (Wang 2009).

$$\begin{aligned} \min_{\Delta U} \left\{ \frac{1}{2} \Delta U^T (\Phi^T \Phi + G) \Delta U + Fx(n) \right\} \\ \text{Subject to } Z \Delta U \leq b \\ \Delta u_{\min} \leq \Delta u \leq \Delta u_{\max} \end{aligned} \quad (18)$$

The constraints may also be used to obtain safety (collision avoidance), comfort and energy efficiency.

The control variable can be limited as expressed in (18), providing a more comfortable driving experience as the vehicle jerk becomes less abrupt. Moreover the optimal solution can be defined in such a way that the resulting relative distance between two vehicles never becomes less than zero. By limiting the relative distance to positive values, we prevent vehicle collision improving the overall safety of the system.

To assure that the computed output does not lead the system to an unsafe state, the prediction horizon of the relative distance must be restricted to values greater or equal to zero. This limitation is achieved by defining $Z = -\Phi$ and the vector $b = Fx(n)$.

4. Results and Discussion

To analyze the performance of the proposed model predictive control, simulations based on Simulink have been carried out by performing tests with different control parameters and constraints.

For the control parameters, it is proposed a sample time of $T_s = 0.1s$ with a prediction horizon of $N_p = 200$ and a control horizon of $N_c = 40$. These values provide a reasonable trade-off between computational burden and control responsiveness.

In this experiment, it was defined two identical vehicles and it was implemented a CACC which control the distance, velocity and acceleration of the first vehicle (vehicle B figure 2) through a jerk control input in order to maintain a safe distance and, at the same time, adjust the velocity to the vehicle in front, in order to avoid collision.

In this simulation, the two vehicles follow at a relative distance of 10m. The front vehicle (vehicle A) follows at a constant velocity of 20 m/s and the controlled vehicle (vehicle B) is initially followed at 18 m/s. In the time instant $t = 0$ s the controller is set and leads the system to a relative distance of 1m and a final velocity of 20 m/s, the same velocity as the vehicle in front.

To visually inspect the influence of the gain G defined in the optimization function (18), the figure 3 shows the evolution of the state variables of the vehicle B, that follows the leader (vehicle A). As it can be seen in figure 3, the state variables converge at a different rate that depends directly on the magnitude of the gain G . The control variable of the system can be seen in figure 4 where it shows the performance of the controller to different gain magnitudes. The control rate is limited to $-2,5 < \Delta U < 2,5$ in this case.

As shown in figure 3, the increase of the magnitude of the gain G makes the system response slower, obtaining a smoother control variable. Another important issue to analyze is the limitation on the control rate ΔU that influence the optimization response. Figure 5 shows a time history for all state variables obtained with different constraints magnitude.

The state variables presented in figure 5 shows that the limitation of the control rate result in a slower response of the system and overshoot. The control variable applied in each case is compared in figure 6. As expected the control variable have a more abrupt characteristic as the limit becomes less restricted.

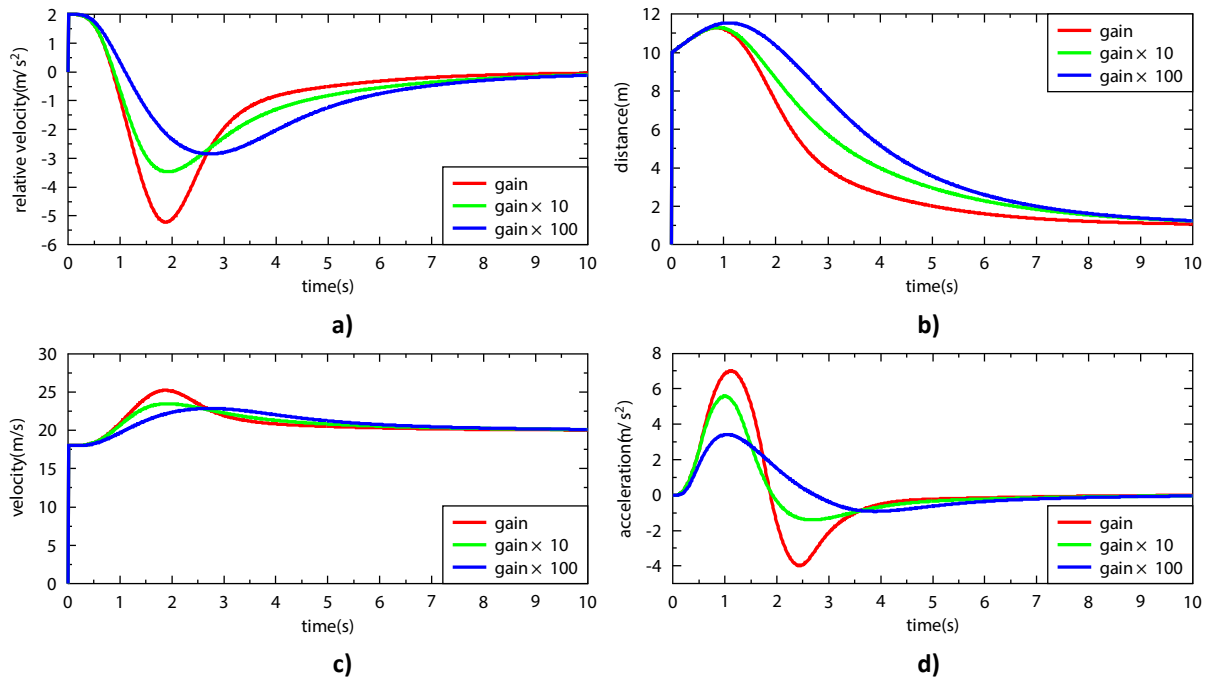


Figure 3: Vehicle B with different gain G **a)** Relative velocity **b)** Relative Distance
c) Absolute velocity **d)** Relative Acceleration

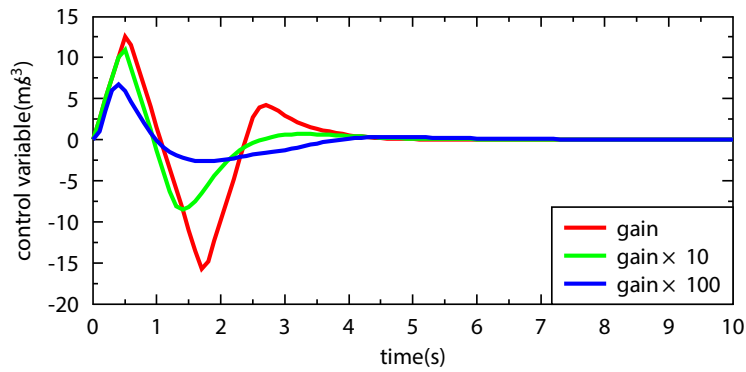


Figure 4: Control Variable of vehicle B with different gain G

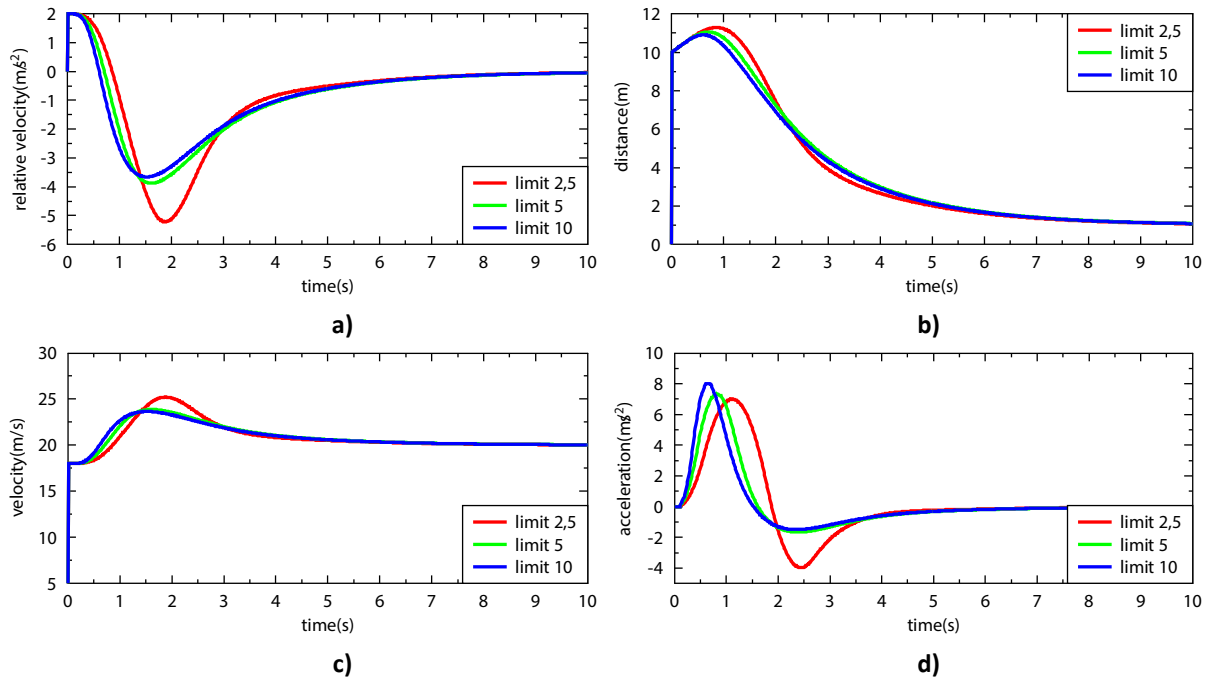


Figure 5: Vehicle B with different control rate limit **a)** Relative velocity **b)** Relative Distance **c)** Absolute velocity **d)** Relative Acceleration

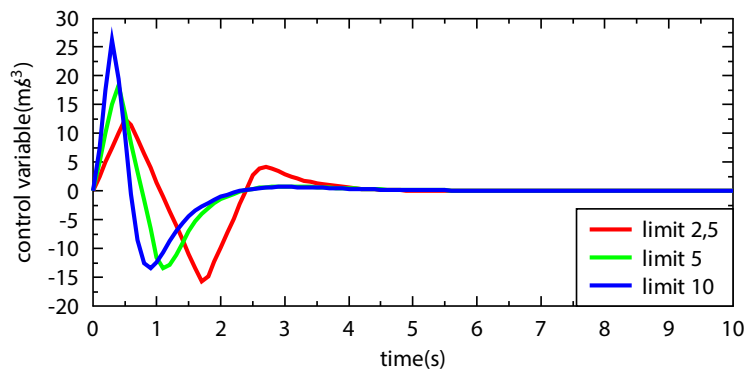


Figure 6: Control Variable of vehicle B with different control rate limit

5. Conclusions

The present paper focuses on MPC in vehicle platoon control. It is assumed that the lower layer of control is based on feedback linearization control law. The MPC approach is used that combines a prediction model with an optimization algorithm which estimate the future states of the system and compute the control action within a control horizon to obtain an optimal solution of the system.

The simulation experiments are performed to test the effectiveness of this control technique. The response of the vehicle is evaluated with different control parameters and limits. From this analysis it becomes clear that the increase of the gain matrix G smooth the control response and may be used as a soft constraint in the system. The hard constraint defined for the control variable produce the desired effect by limiting the control rate, obtaining a different control dynamic. The other hard constraint implemented to limit the system relative distance error to positive values and prevents the system to overshoot promoting a safe system and ensuring that a vehicle will not slip to an unsafe region, avoiding the collision with other vehicles.

In future work the controller can be further improved through the introduction of other control constraints. The inclusion of other vehicles in the platoon will also be considered. Another research topic will study the string stability of the system.

References

- Dey, K. C., L. Yan, X. Wang, Y. Wang, H. Shen, M. Chowdhury, L. Yu, C. Qiu, and V. Soundararaj. 2016. "A review of communication, driver characteristics, and controls aspects of cooperative adaptive cruise control (CACC)." *IEEE Transactions on Intelligent Transportation Systems* no. 17 (2):491-509. Accessed May 11, 2016. DOI: [10.1109/TITS.2015.2483063](https://doi.org/10.1109/TITS.2015.2483063).
- Girard, A. P., S. Spry, and J. K. Hedrick. 2005. "Intelligent cruise control applications: real-time embedded hybrid control software." *IEEE Robotics & Automation Magazine* no. 12 (1):22-28. Accessed May 11, 2016. DOI: [10.1109/MRA.2005.1411415](https://doi.org/10.1109/MRA.2005.1411415).
- Girard, A. R., J. B. de Sousa, J. A. Misener, and J. K. Hedrick. 2001. "A control architecture for integrated cooperative cruise control and collision warning systems." *Proceedings of the 40th IEEE Conference on Decision and Control, 2001* no. 2:1491-1496 vol.2. Accessed May 11, 2016. DOI: [10.1109/.2001.981105](https://doi.org/10.1109/.2001.981105).
- Godbole, D. N., and J. Lygeros. 1994. "Longitudinal control of the lead car of a platoon." *IEEE Transactions on Vehicular Technology* no. 43 (4):1125-1135. Accessed May 11, 2016. DOI: [10.1109/25.330177](https://doi.org/10.1109/25.330177).
- Luo, Li-hua, Hong Liu, Ping Li, and Hui Wang. 2010. "Model predictive control for adaptive cruise control with multi-objectives: comfort, fuel-economy, safety and car-following." *Journal of Zhejiang University SCIENCE A* no. 11 (3):191-201. Accessed May 11, 2016. DOI: [10.1631/jzus.A0900374](https://doi.org/10.1631/jzus.A0900374).
- Shakouri, Payman, and Andrzej Ordys. 2014. "Nonlinear model predictive control approach in design of adaptive cruise control with automated switching to cruise control." *Control Engineering Practice* no. 26:160-177. Accessed May 11, 2016. DOI: [10.1016/j.conengprac.2014.01.016](https://doi.org/10.1016/j.conengprac.2014.01.016).
- Shakouri, P., A. Ordys, D. S. Laila, and M. Askari. 2011. "Adaptive cruise control system: comparing gain-scheduling PI and LQ controllers." *IFAC Proceedings Volumes* no. 44 (1):12964-12969. Accessed May 11, 2016. DOI: [10.3182/20110828-6-IT-1002.02250](https://doi.org/10.3182/20110828-6-IT-1002.02250).
- Swaroop, D., and J. K. Hedrick. 1996. "String stability of interconnected systems." *IEEE Transactions on Automatic Control* no. 41 (3):349-357. Accessed May 11, 2016. DOI: [10.1109/9.486636](https://doi.org/10.1109/9.486636).
- Swaroop, D., J. K. Hedrick, and S. B. Choi. 2001. "Direct adaptive longitudinal control of vehicle platoons." *IEEE Transactions on Vehicular Technology* no. 50 (1):150-161. Accessed May 11, 2016. DOI: [10.1109/25.917908](https://doi.org/10.1109/25.917908).
- Vahidi, A., and A. Eskandarian. 2003. "Research advances in intelligent collision avoidance and adaptive cruise control." *IEEE Transactions on Intelligent Transportation Systems* no. 4 (3):143-153. Accessed May 11, 2016. DOI: [10.1109/TITS.2003.821292](https://doi.org/10.1109/TITS.2003.821292).
- van Arem, B., C. J. G. van Driel, and R. Visser. 2006. "The impact of cooperative adaptive cruise control on traffic-flow characteristics." *IEEE Transactions on Intelligent Transportation Systems* no. 7 (4):429-436. Accessed May 11, 2016. DOI: [10.1109/TITS.2006.884615](https://doi.org/10.1109/TITS.2006.884615).
- Vinel, A., L. Lan, and N. Lyamin. 2015. "Vehicle-to-vehicle communication in C-ACC/platooning scenarios." *IEEE Communications Magazine* no. 53 (8):192-197. Accessed May 11, 2016. DOI: [10.1109/MCOM.2015.7180527](https://doi.org/10.1109/MCOM.2015.7180527).

Wang, Liuping. 2009. *Model predictive control system design and implementation using MATLAB®*. London: Springer London. Accessed May 11, 2016. DOI: [10.1007/978-1-84882-331-0](https://doi.org/10.1007/978-1-84882-331-0).