

This is the peer reviewed version of the following article:

A control matching model predictive control approach to string stable vehicle platooning / Kianfar, Roozbeh; Falcone, Paolo; Fredriksson, Jonas. - In: CONTROL ENGINEERING PRACTICE. - ISSN 0967-0661. - 45:(2015), pp. 163-173. [10.1016/j.conengprac.2015.09.011]

Terms of use:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

03/05/2026 05:07

(Article begins on next page)

A Control Matching Model Predictive Control Approach to String Stable Vehicle Platooning

Roosbeh Kianfar, Paolo Falcone, Jonas Fredriksson

Department of Signals and Systems, Chalmers University of Technology, SE-412 96 Göteborg, Sweden, (e-mail: roosbeh.paolo.falcone.jonas.fredriksson@chalmers.se).

Abstract

A predictive control strategy for vehicle platoons is presented in this paper, accommodating both string stability and constraints (e.g., physical and safety) satisfaction. In the proposed design procedure, the two objectives are achieved by matching a Model Predictive Controller (MPC), enforcing constraints satisfaction, with a linear controller designed to guarantee string stability. The proposed approach neatly combines the straightforward design of a string stable controller in the frequency domain, where a considerable number of approaches have been proposed in literature, with the capability of an MPC-based controller enforcing state and input constraints.

A controller obtained with the proposed design procedure is validated both in simulations and in the field test, showing how string stability and constraints satisfaction can be simultaneously achieved with a single controller. The operating region that the MPC controller is string stable is characterized by the interior of feasible set of the MPC controller.

Keywords: Model Predictive Control, vehicle platooning, string stability, control matching

1. Introduction

Automated highways, in particular vehicle platooning, is considered as an appealing technology to contribute alleviating traffic flow problem like congestions.

5 A vehicle platoon consists of a chain of automated vehicles following each other led by a specific vehicle, i.e., the leader. The primary objective of a platoon is that vehicles should follow each other by maintaining a desired gap/distance to their preceding vehicles. The
10 idea of platooning dates back to the eighties when California Partner for Advanced Transportation Technology (PATH) was established to study and develop intelligent vehicle-highway cooperation and communication systems, [1]. Since then, several studies on the potential
15 impact of vehicle platooning on different aspect of transportation and traffic flow have been conducted, [2]. Potential benefit of vehicle platooning in reducing the fuel consumption is studied in [3]. The impact of vehicle platooning on the traffic flow is studied in e.g., [4],
20 [5] and [6]. Regardless of the business model, to enable vehicle platooning, controllers must be developed to maintain a desired distance/time gap between vehicles.

In platooning, the longitudinal dynamics are controlled

25 relying on measurements from on board sensors, e.g. radar and camera. However, to enable a short inter-vehicles distance between vehicles and to guarantee the so called *string stability* property, wireless communication may be required as well [7], [8].

30 The main challenges in the design of a longitudinal dynamics controller for vehicle platooning applications are *i*) satisfying safety and performance requirements within the actuators limitations (hereafter refer to as time domain requirements) *ii*) guaranteeing string stability. As it is shown in previous works, [9, 10, 11], control specifications and requirements, including safety,
35 performance and actuators limitations can be formulated as inequality constraints in a model predictive control formulation. Alternatively, constraints satisfactions and safety can be verified a posteriori for any linear controller by using the set based approaches as in, e.g., [12] and [13].

40 The focus of this paper is the design of a controller accommodating both time domain requirements and string stability. String stability is defined as the ability of a vehicle platoon to attenuate the effect of disturbances introduced by the leader (or any other vehicles) as it propagates down stream in the platoon. However, slightly

different definitions for string stability can be found in the literature, e.g., string stability w.r.t. different disturbance signals and different norm sense [14]. In this work, string stability is defined as the capability of a vehicle platoon in attenuating the energy of the acceleration signals as moving toward the tail of the platoon, as proposed by [15]. However, it should be noted that string stability cannot guarantee safety. This motivates the fact that controllers should be equipped with a tool such that they can handle constraints (e.g., safety) explicitly.

Alternative designs of string stable vehicle platoons in the frequency domain are given in, e.g., [16], [17], [18] and [19].

Combining string stability and constraints satisfaction requirements in a single controller is not trivial. In general, guaranteeing constraints satisfaction, is not trivial in frequency domain designs. On the other hand translating the frequency domain definition of string stability into time domain settings as MPC, is not trivial either. In [20] and [21], the string stability requirement is translated into inequality constraints in an MPC controller. However, the proposed methods require that each vehicle broadcasts an intended trajectory to its followers which might be impractical with **couldn't read your comment** wireless communication technology.

In this work, we propose a predictive control design procedure for vehicle platoons, accommodating both string stability and constraints (e.g., physical and safety) satisfaction. This is a two-step procedure. In the first step, a linear controller is designed in order to guarantee string stability. It is important to point out that the design can be based on any string stability definition and by resorting to any design procedure leading to a linear control structure. In this paper, as an example, a controller based on $H_{\infty/2}$ is designed to guarantee string stability. The choice of an H_{∞} controller is well motivated considering the \mathcal{L}_2 string stability criterion adopted in this work. However, any other linear controller which can result in a string stable vehicle platoon can be suitable as well. Then, in the second step, the control problem is formulated in an MPC framework with the ability of handling the time domain constraints. Furthermore, a controller matching approach is used to tune the weighting matrices of the MPC controller such that its behavior matches the string stable controller while the constraints are not active, [22]. In particular, the physical, safety and design constraints are embedded in an MPC controller. Then a convex optimization problem is solved to find the weighting matrices of MPC controller such that the same behavior as the string stable controller achieved by MPC controller. Accomplishing the afore-

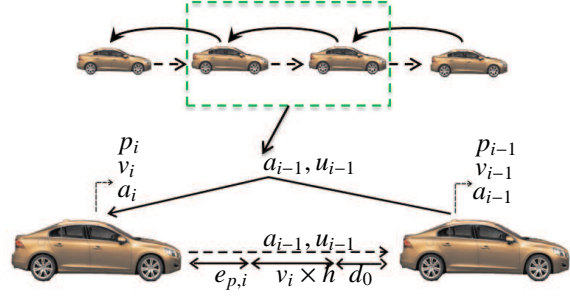


Figure 1: Two adjacent vehicles in the platoon.

mentioned two steps results in a string stable MPC controller with the capability of fulfilling constraints, e.g. safety and actuator limitation.

2. Vehicle Modeling

Consider two adjacent vehicles, as shown in Fig. 1. Let p_{i-1} , v_{i-1} and a_{i-1} denote the position, velocity and acceleration of the vehicle preceding the i -th vehicle (ego vehicle) in a platoon and p_i , v_i and a_i denote the position, velocity and acceleration of the i -th vehicle. Denote by e_p the position error w.r.t. a desired distance from the preceding vehicle, i.e., $e_{p,i} = p_{i-1} - p_i - d_0 - v_i h_i$, where d_0 and h_i are a constant safety distance and the constant headway time, respectively. The headway time is the time necessary to the ego vehicle to travel the distance to the preceding vehicle, at its current speed. The error dynamics are then described by the following set of equations

$$\begin{aligned} \dot{e}_{p,i} &= e_{v,i} - a_i h_i, \\ \dot{e}_{v,i} &= a_{i-1} - a_i, \end{aligned} \quad (1)$$

where $e_{v,i}$ is the relative velocity. The longitudinal acceleration dynamics can be described by,

$$a_i = \frac{K_i}{\tau_i s + 1} e^{-\theta_i s} a_i^{\text{des}}, \quad (2)$$

where K_i , τ_i and θ_i are the steady state gain, the time constant of the actuator (engine and brake) and the actuator delay, respectively and a_i^{des} is the demanded acceleration, [23]. The model (1)-(2) can then be written in a state-space form as

$$\dot{x}(t) = Ax(t) + B_u u(t - \theta) + B_\omega \omega(t), \quad (3)$$

105 where

$$A = \begin{bmatrix} 0 & 1 & -h_{i+1} & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & -1/\tau_{i+1} & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad (4)$$

$$B_u = \begin{bmatrix} 0 \\ 0 \\ \frac{K_{i+1}}{\tau_{i+1}} \\ 0 \end{bmatrix}, B_\omega = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \quad (5)$$

and

$$x = \begin{bmatrix} e_{p,i} & e_{v,i} & a_i & v_i \end{bmatrix}^T, \quad (6)$$

$$u_i = a_i^{\text{des}}, \quad (7)$$

$$\omega = a_{i-1}, \quad (8)$$

are the state, control and disturbance vectors, respectively. Notice that the acceleration of the preceding vehicle is considered as a measured disturbance.

110 3. Constraints and time domain requirements

Control objective is to minimize the position and velocity errors while satisfying a number of requirements described next. The requirements are written for a vehicle i .

115 3.1. Safety:

The safety requirement is introduced to guarantee that a safe *minimum* distance is maintained from the preceding vehicle in order to prevent rear-end collisions. Based on the notation introduced in Section 2, the safety requirement can be written as

$$e_{p,\min} \leq e_{p,i}(t) \leq e_{p,\max}, \quad \forall t \geq 0, \quad (9)$$

where $e_{p,\max}$ is the maximum allowed distance from the preceding vehicle. We observe that, $e_{p,\max}$ can be selected according to performance criteria (e.g., to not allow increasing the platoon length), the choice of lower bound in (9) $e_{p,\min}$ is based on safety requirement.

3.2. Performance:

Since the primary objective of the automated driving system is to regulate the vehicle velocity to the platoon velocity, the relative speed between the two adjacent vehicles is constrained as,

$$e_{v,\min} \leq e_{v,i}(t) \leq e_{v,\max}, \quad \forall t \geq 0, \quad (10)$$

where $e_{v,\min}$ and $e_{v,\max}$ can be chosen based on the performance requirement.

3.3. Acceleration requirement

To ensure that all vehicles belonging to a platoon deliver the same amount of acceleration/deceleration, based on the platoon requirement, the following constraint can be used

$$a_{\min} \leq a_i(t) \leq a_{\max}, \quad \forall t \geq 0, \quad (11)$$

125 where a_{\min} and a_{\max} , are the minimum and maximum allowed acceleration, respectively.

3.4. Actuator limitations:

Due to the physical limitation of actuator (engine and brake), acceleration commanded by the controller is limited by the the following constraints:

$$u_{\min} \leq u_i(t) \leq u_{\max}, \quad \forall t \geq 0. \quad (12)$$

3.5. Desired velocity range:

According to the platoon requirements, the vehicles should be able to operate within a desired velocity range which can be set through the following constraint:

$$0 \leq v_i(t) \leq v_{\max}, \quad \forall t \geq 0, \quad (13)$$

where v_{\max} is the maximum allowed velocity. Constraints (9)-(13) can be compactly rewritten in the following form,

$$\begin{bmatrix} H_x & H_u \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \leq h_x. \quad (14)$$

where (14) is a polytope in the input and state space. The projections of (14) onto the state space and input space are represented by \mathcal{X} and \mathcal{U} , respectively.

3.6. String stability:

String stability is an important property of a vehicle platoon which should be considered in controller design. Among alternative definitions of string stability, we adopt a predecessor-follower string stability criterion based on the \mathcal{L}_2 induced norm w.r.t. to acceleration signals, [24]. Denote by a_{i-1} and a_i the acceleration signals of two adjacent vehicles. Then string stability is defined as,

Definition 1 (String stability). A vehicle platoon is predecessor-follower string stable w.r.t. a acceleration disturbance if the following holds,

$$\|\Gamma_i(s)\|_{\infty} = \sup_{\omega \in \mathbb{R}} \sigma(\Gamma_i(j\omega)) \leq 1, \quad (15)$$

where $\Gamma_i(s)$ and σ are the transfer function from $a_{i-1}(s) \rightarrow a_i(s)$ and the singular value, respectively. This condition can be written in terms of \mathcal{L}_2 induced norm of input output,

$$\|\Gamma_i(s)\|_\infty = \sup_{a_i \neq 0} \frac{\|a_i(t)\|_{\mathcal{L}_2}}{\|a_{i-1}(t)\|_{\mathcal{L}_2}}. \quad (16)$$

Condition (15) states that the total energy of output signal is less than the total energy of input signal over interval $t \in [0, \infty)$.

Remark 1. The presented predecessor-follower string stability criterion is more stringent than the leader-follower string stability, [20].

Remark 2. An alternative definition for string stability can be defined using \mathcal{L}_∞ , which can guarantee no overshoot w.r.t. to desired signals while the signal propagates in the platoon [16].

4. Control problem formulation

In this work a control matching MPC strategy is adopted which simultaneously can fulfill specifications, requirements and string stability condition presented in Section 3. In this section, we present the two necessary steps to design the controller. First, a mixed $H_\infty/2$ is synthesized to guarantee string stability condition (16). Then control problem is formulated in an MPC framework to fulfill constraints (9)-(12). The weighting matrices in the objective function of MPC controller are chosen according to the control matching technique proposed in [22]. Details on the control design are presented in the sequel.

4.1. String stable controller

Consider the feedback/feedforward control policy,

$$u_{ss,i} = K^{ss} \begin{bmatrix} x \\ w \end{bmatrix} = \begin{bmatrix} K_{FB}^{ss} & K_{FF}^{ss} \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix}, \quad (17)$$

where K_{FB}^{ss} and K_{FF}^{ss} are static state feedback and feedforward gains, respectively. Utilizing the control policy (17) and setting $u = u_{ss,i}$ in (3), yields the closed loop system,

$$\dot{x}(t) = A^{cl}x(t) + E_w^{cl}w(t), \quad (18)$$

where $A^{cl} = A + B_uK_{FB}^{ss}$ and $E_w^{cl} = B_uK_{FF}^{ss} + E_w$. We consider the following output signals,

$$\begin{bmatrix} a_i \\ e_{p,i} \\ u_{ss,i} \end{bmatrix} = \begin{bmatrix} C_\Gamma & D_\Gamma \\ C_H & D_H \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix}, \quad (19)$$

and denote by, $\Gamma = C_\Gamma (sI - A^{cl})^{-1} E_w^{cl} + D_\Gamma$, the transfer function from w to a_i and by $H = C_H (sI - A^{cl})^{-1} E_w^{cl} + D_H$, the transfer function from w to $[e_{p,i} \ u_{ss,i}]^T$.

Naturally, we want to choose the gain K^{ss} in (17) to stabilize position error $e_{p,i}$. In addition we would like to keep the control effort and acceleration as small as possible. In the frequency domain we thus want to chose K^{ss} such that, the 2-norm, $\|H\|_2$ is minimized.

We also want to find K^{ss} such that acceleration signals are attenuated as much as possible along the tail of the platoon. We thus want to optimize K^{ss} such that $\|\Gamma\|_\infty$ is minimized. In particular the condition (15) should be satisfied in order to ensure string stability. We calculate K^{ss} as,

$$K^{ss} = \underset{K^{ss}}{\operatorname{argmin}} \quad \alpha\|\Gamma\|_\infty + \beta\|H\|_2, \quad (20a)$$

$$\text{subj.to} \quad \|\Gamma\|_\infty \leq 1, \quad (20b)$$

where α and β are non-negative tuning parameters that can be chosen to trade off between the different control objectives. The optimization problem (20) can be formulated as an LMI which can be solved using convex optimization solvers [25].

4.2. Model Predictive controller

In order to stabilize the error dynamics with a receding horizon framework, the system dynamics (3) are discretized using the Euler method with sampling time t_s :

$$x(t+1) = F_i x(t) + G_{1,i} u(t) + G_{2,i} \omega(t). \quad (21)$$

We recall that the control objective of each vehicle is to regulate the position and velocity errors e_p and e_v to zero, respectively, while satisfying the constraints described in Section 3.

We assume that the state and the disturbance vectors can be measured every sampling time instant t_s , and solve the following quadratic programming (QP) optimization problem in receding horizon,

$$J(x(t), \omega(t)) = \min_{U_t} \|Px(N|t)\|_2 \quad (22a)$$

$$+ \sum_{k=0}^{N-1} \|Qx(t+k|t)\|_2 + \|Ru(t+k|t)\|_2 \quad (22b)$$

$$\text{subj. to } x_{t+k+1,t} = F_i x_{t+k,t} + G_{1,i} u_{t+k,t} + G_{2,i} \omega_{t+k,t}, \quad (22c)$$

$$\begin{bmatrix} H_x & H_u \end{bmatrix} \begin{bmatrix} x_{t+k,t} \\ u_{t+k,t} \end{bmatrix} \leq h_x, \quad (22d)$$

$$x_{t+N,t} \in \mathcal{X}_f = \mathbb{R}^n \quad (22e)$$

$$\omega_{t+k|t} = \omega_{t|t} \quad (22f)$$

$$k = [0, \dots, N-1],$$

$$\omega_{t,t} = \omega(t), \quad (22g)$$

$$x_{t,t} = x(t), \quad (22h)$$

190 where $U_t = [u(t), \dots, u(t+N-1)] \in \mathbb{R}^N$ is the vector of input signal over the horizon, i.e., the vector of optimization variables, N is the prediction horizon length, $Q \geq 0$, $R > 0$ and $P \geq 0$ are weighting matrices of appropriate dimensions on state, control signal and final state, respectively. Constraints (22d) include the safety, performance requirements and actuators limitations. Note that to reject the measured disturbance ω , the prediction model (22c) can be written in the augmented form where the augmented state vector is $\tilde{x} = [x, \omega]^T$, while the dynamic of $\omega_{t+k|t}$ is described by (22f).

4.2.1. Control Matching Problem

As shown in (22), every time instant a Quadratic Programming problem (QP) is solved. Given the initial conditions, $x(k)$ and $\omega(k)$, the QP-problem can be written as follows,

$$J(x(k), \omega(k), U(k)) = \min_{U(k)} U(k)^T H U(k) + 2x(k)^T F U(k) + x(k)^T Y x(k) \quad (23a)$$

subj. to

$$M U(k) \leq W(k) + E x(k), \quad (23b)$$

the weighting matrices in the objective function (23a), H, F, Y and polyhedral constraint (23b) with $M \in \mathbb{R}^{r \times N}$, $E \in \mathbb{R}^{r \times n}$ and $W \in \mathbb{R}^r$ are defined as follows,

$$\begin{aligned} H &= (S^u \bar{Q} S^u + \bar{R}), & F &= S^x \bar{Q} S^u, \\ Y &= S^x \bar{Q} S^x \end{aligned} \quad (24)$$

where S^u and S^x are,

$$S^u = \begin{bmatrix} F & 0 & \cdots & 0 \\ FG1 & G1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ F^{N-1}G1 & F^{N-2}G1 & \cdots & G1 \end{bmatrix}, S^x = \begin{bmatrix} F \\ F^2 \\ \vdots \\ F^N \end{bmatrix}, \quad (25)$$

and \bar{Q}, \bar{R} are block diagonal matrices,

$$\bar{Q} = \begin{bmatrix} Q_1 & 0 & 0 & \cdots & 0 \\ 0 & Q_2 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & Q_{N-1} & 0 \\ 0 & 0 & \cdots & 0 & P \end{bmatrix}, \quad (26)$$

$$\bar{R} = \begin{bmatrix} R_0 & 0 & \cdots & 0 \\ 0 & R_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R_{N-1} \end{bmatrix}, \quad (27)$$

The solution to the unconstrained optimization problem (23a), that is, when the constraints (23b) are not active, i.e., $M U(k) < W(k) + E x(k)$ can be obtained by using the necessary and sufficient optimality condition, Hence the vector of optimal control sequence $U^*(k)$ can be calculated as follows,

$$U^*(k) = -H^{-1} F^T x(k). \quad (28)$$

The first element of $U^*(k)$ can be extracted as following and is applied to the plant at time k .

$$u^*(k) = u(0|k) = -\Lambda H^{-1} F^T x(k), \quad (29)$$

where $\Lambda = [I \ 0 \ \cdots \ 0]$.

The control matching problem can be seen as the problem of calculating the weighting matrices of the optimization problem (22), i.e., $Q \geq 0$, $P \geq 0$ and $R > 0$ such that the control action calculated by the MPC controller, $u^*(k)$ in (29) is equal to the control action calculated by the string stable controller, u_{ss} in (20a). This amounts to solving the following,

$$u_{ss}(k) = [K_{FB} K_{FF}] \tilde{x}(k) = -\Lambda H^{-1} F^T \tilde{x}(k) \quad (30)$$

where $\tilde{x}(k) = [x(k), \omega(k)]^T$ is the augmented state vector. The variable to be found in (30) are Q, P and R according to (24)-(27). Solving (30) for Q, P and R requires inverting the singular matrix Λ . To remove Λ from (30), we enforce the solution to the unconstrained MPC problem be the same as the favourite controller over an N -step prediction horizon instead,

$$[k_0, k_1, \dots, k_{N-1}]^T \tilde{x}(k) = -H^{-1} F^T \tilde{x}(k). \quad (31)$$

To enforce that the first control action of MPC controller matches the string stable $H_{\infty/2}$ controller, we set $k_0 = [K_{FB}K_{FF}]$, which the gain k_i for $i \in [1, N-1]$ can be chosen freely. Equation (31) can be cast as an optimization (or feasibility) problem where the weighting matrices Q , P , R and free matrix gains k_i are the optimization variables. However, considering k_i as free variables results in an optimization problem subject to bilinear matrix inequalities (BMI). Even though, nowadays, there are available solvers to tackle such problem, BMI problems are still considered as NP hard problem due to their non-convex nature. Assuming that the matrix gains k_i are fixed (pre-specified), (31) can be cast as a semi definite programming (SDP), i.e. linear objective function subject to LMI, [22],

Lemma 1. Consider the following convex optimization problem

$$\mathcal{V} = \min_{Q_i, P, R_i} \|(\bar{R} + S^u \bar{Q} S^u) \mathcal{K} + S^u \bar{Q} S^u\| \quad (32a)$$

$$\text{subj. to } P \geq 0, \quad R_i \geq \sigma I, \quad i = 0, \dots, N-1 \\ Q_i \geq 0, \quad i = 1, \dots, N-1 \quad (32b)$$

where \mathcal{K} is the matrix of pre-specified gains over the N -step prediction horizon,

$$\mathcal{K} = \begin{bmatrix} K \\ K(A+BK) \\ \vdots \\ K(A+BK)^{N-1} \end{bmatrix}. \quad (33)$$

Clearly, if the objective function (32a) becomes zero at the optimum, i.e., $\mathcal{V}^* = 0$, then the solution to the optimization problem, i.e. Q_i , P and R_i are also the exact solution to the (31). Hence, if $\mathcal{V}^* = 0$, the MPC controller (22) will behave the same as the string stable controller presented in Section 4.1, as long as the constraint (23b) are not active.

Note that the MPC controller proposed in Section 4.2 can enforce string stability as long as the constraints are not active, i.e. when the following holds,

$$MU(k) < W(k) + Ex(k) \quad (34)$$

This corresponds to the interior of feasible (admissible) set of the MPC controller. Once the constraints are active, the behavior of MPC controller would be different than the string stable controller in general. However, this is not restrictive since when the constraints are active, e.g. either the safety is endanger or the control signal is saturated. Clearly, if the safety is endanger, then string stability is not the priority of the controller, and if the control signal is saturated the string stability cannot be guaranteed with string stable controller neither.

5. Feasibility and Stability

In general feasibility and global stability of an MPC controller is not guaranteed. In particular for the control matching technique utilized in the previous section, persistent feasibility and global stability can be guaranteed by introducing terminal constraint and penalty on the final states.

5.1. Preliminaries

First, a few definitions are introduced which are used later to guarantee persistent feasibility and global stability of MPC controller (22).

Definition 2 (Robust backward reachable set).

Denote by \mathcal{T} as a desired target set, the robust backward reachable set with polytopic uncertainty for system (6) subject to the constraint (14) defines as,

$$\text{Pre}(\mathcal{T}, \mathcal{W}) = \{x(t) \in \mathbb{R}^n : \forall \omega(t) \in \mathcal{W}, \\ \exists u(t) \in \mathcal{U} | Ax(t) + E\omega(t) \in \mathcal{T}\}, \quad (35)$$

the set of states which by using a feasible control signal $u \in \mathcal{U}$ can evolve to the target set \mathcal{T} for all $\forall \omega \in \mathcal{W}$.

For the closed loop system (3, 29) subject to the constraint (14) the robust backward reachable set is defined as,

$$\text{Pre}(\mathcal{T}, \mathcal{W}) = \\ \{x(t) \in \mathbb{R}^n : \forall \omega(t) \in \mathcal{W}, |Ax(t) + E\omega(t) \in \mathcal{T}\}, \quad (36)$$

the set of states which can evolve to the target set \mathcal{T} for $\forall \omega \in \mathcal{W}$.

Definition 3 (Positive invariant set). A set $O \subset \mathcal{X}$ is said to be positive invariant set for the closed loop system (3, 29) subject to the constraint (14), if

$$x(0) \in O \Rightarrow x(t) \in O \quad \forall \omega(t) \in \mathcal{W}, \forall t \geq 0, \quad (37)$$

the set $O_{\infty} \subset \mathcal{X}$ is said to be the maximal positive invariant set, if it is positive invariant set and contains all positive invariant set contained in \mathcal{X} .

Definition 4 (Control invariant set). A set $C \subset \mathcal{X}$ is said to be control invariant set for the system (6) subject to the constraint (14), if

$$x(t) \in C \Rightarrow \exists u(t) \in \mathcal{U} \quad \text{s.t} \\ Ax(t) + Bu(t) + E\omega(t) \in C, \forall \omega(t) \in \mathcal{W}, \forall t \geq 0, \quad (38)$$

the set $C_{\infty} \subset \mathcal{X}$ is said to be the maximal control invariant set, if it is control invariant set and contains all control invariant set contained in \mathcal{X} .

5.2. Persistent feasibility

Denote by \mathcal{X}_f the terminal constraint for the MPC controller (22). To obtain persistent feasibility, i.e., feasibility for all future time, the terminal constraint (22e) in MPC problem (22) can be set as a control invariant set. The details are given in the following lemma.

Lemma 2. Consider the MPC controller (22) with $N \geq 1$. If \mathcal{X}_f is a control invariant set for system (6) then the MPC controller is persistently feasible.

Details of proof can be found in [].

The recursive algorithm for calculating the positive and control invariant set is given in Algorithm 1. Essentially, calculating both positive and control invariant set requires the same steps. However, for the control and positive invariant set the robust backward reachable set is calculated based on (37) and (38), respectively.

Algorithm 1 Maximal positive/control invariant set

Let $\Omega = \{O, C\}$
 $i = 0,$
 $\Omega_i = \mathcal{X},$
repeat
 $i = i + 1$
 $\Omega_{i+1} = \text{Pre}(\Omega_i, \mathcal{W}) \cap \Omega_i$
until $\Omega_{i+1} == \Omega_i$
 $\Omega_\infty = \Omega_i$

In Fig. 2, the maximal control invariant set of system (6) subject to the constraint (14) is depicted in yellow. The constraints on position error e_p and e_v are set to be $-3 \leq e_p \leq 3$ and $-4 \leq e_v \leq 4$, respectively. The positive invariant set for the system (3) stabilized with $u(k) = K_{FB}x(k)$ subject to the constraint (14) is shown in dark yellow.

Remark 3. As discussed earlier, introduction of terminal constraints can result in a persistently feasible MPC controller. However, it should be noted that this is true as long as the disturbance is within the pre-assumed set \mathcal{W} . In case of an unmodelled disturbance or a model mismatch infeasibility still may occur. To overcome this problem the constraints are softened by introducing a slack variable $\epsilon \geq 0$ which is quadratically penalized in the objective function (22).

5.3. Global stability

Apparently, stability of MPC controller is guaranteed as long as the constraints are not active due to the stability of string stable controller. However, to guarantee the global stability of MPC controller we can impose

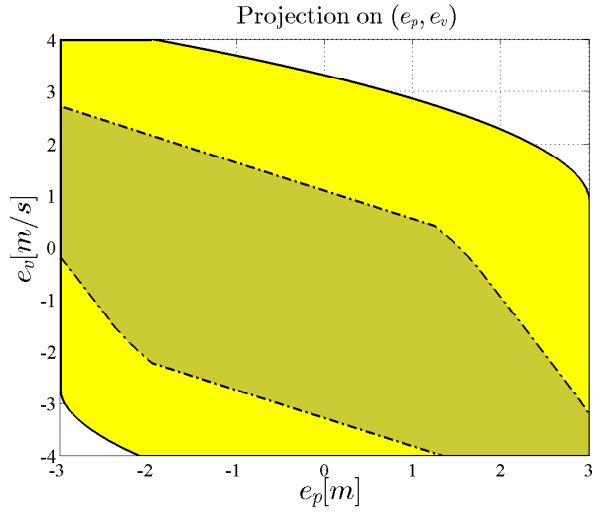


Figure 2: Maximal control invariant set C_∞ and maximal positive invariant set O_∞ are in light and dark yellow, respectively.

constraint on the terminal states (22e) and also penalizing the to final states in the objective function (22).

Lemma 3. Consider the MPC controller (22), introducing the terminal constraint $\mathcal{X}_f = O_\infty$ where O_∞ is the positive invariant set for the closed loop system $x(k+1) = (A + BK_{FB})x(k)$ with K_{FB} a stabilizing controller for the system (3). By introducing the following LMI constraint to the optimization problem (32a)

$$(A + BK_{FB})^T P (A + BK_{FB}) + K_{FB}^T R K_{FB} + Q - P \leq 0, \quad (39)$$

the weighting matrices Q , P and R are found such that the resulting closed loop MPC dynamics are asymptotically stable.

For details on the proof of stability of the control matching technique, we refer to [22].

6. Simulations and Experimental Results

In this study a platoon of three vehicles is considered to evaluate the performance and effectiveness of the proposed method in simulations and experiments, respectively. Simulations and experiments are used to demonstrate the capability of the controller proposed in Section 4. In particular, three scenarios are considered to evaluate the ability of platoon in *i*) following preceding vehicle with constant time gap *ii*) maintaining string stability (16) and *iii*) fulfilling constraints (14). Experiments are performed in a straight track of almost 400m length. Simulation and experiments scenarios are detailed in the sequel.

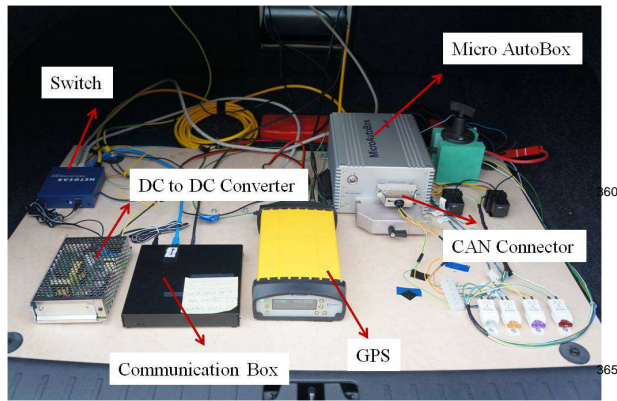


Figure 3: The truck of one of the following vehicles.

6.1. Description of the Experimental Set-up

The experiments were performed at Chalmers University in a three vehicle platoon, with a leader and two followers. The lead vehicle is a Volvo S80, equipped with

- a Ublox EVK-5H GPS evaluation kit, based on a Ublox LEA-5H GPS receiver module, updating the position with an accuracy of 2.5 m at a maximum frequency of 4 Hz ,
- a communication box based on a PC Engines Alix3d2 board, a Mikrotik 802.11a/b/g wireless MiniPCI card with an Atheros AR5414 chipset and an output power up to 350 mW and a 6 dBi radio antenna,
- an ethernet gateway, forwarding signals from the vehicle CAN bus to a Local Area Network,
- a notebook forwarding signals, sent by the GPS through a USB interface, to the LAN,
- an ethernet switch.

The signals from the CAN bus that are broadcast, together with the vehicle global position and the corresponding timestamp, are the vehicle longitudinal speed and acceleration and the yaw rate. The vehicle is driven manually. I.e., a driver is instructed to follow a given speed profile.

The two followers are Volvo S60 and, as shown in Figure 3, are identically equipped with

- a Trimble GPS receiver SPS85, updating the position with a maximum frequency of 20 Hz and an accuracy of $< 1\text{ m}$ and interfaced to the rest of the system through an ethernet port,

- the same communication box as the lead vehicle,
- a dSpace MicroAutoBox II, a rapid prototyping system based on an IBM PowerPC processor running at 900 MHz ,
- an ethernet switch.

Communication is established among the three vehicles through the protocol IEEE 802.11p, implemented in a OpenWrt environment. This is a Linux distribution for embedded systems. Follower $i - 1$ and i , fuse the information from their onboard sensors and GPSs with the information shared via V2V communication through standard Kalman filters outlined in [11], to obtain their relative positions and velocities within the platoon

6.1.1. Vehicle model identification

To identify the parameters in the model (2) experiments are performed. In such experiments a step command is sent to the vehicle and the acceleration response of the vehicle is measured. The experiment is done for acceleration and deceleration maneuver. The step responses can be seen in Fig. (4(a)),(4(b)). The step command is plotted in dashed red, the acceleration measurements and the simulated output of estimated model are shown in dashed dotted black and solid blue, respectively. Using these experiments the parameters, i.e., static gain K_i , time constant τ_i and the actuator time delay θ_i in longitudinal vehicle dynamics (2) can be estimated according to the Fig. (4(a)),(4(b)).

6.2. Scenario 1 (vehicle following)

The primary objective of a vehicle platoon is that each vehicle should be able to adjust its velocity to the velocity of the leader while following its preceding vehicle with a desired distance. To demonstrate the capability of a platoon equipped with controllers proposed in Section (4) an experiment with three aforementioned prototype vehicles is conducted. The experiment start at a very low speed and the the lead vehicle start accelerating to reach a speed of about 10 m/s . The lead vehicle keeps the speed constant for about 25 s and then start braking to almost standstill. As can be seen from Fig. 5, the vehicles in the platoon are capable of following each other at the desired speed with almost no position errors at the steady state. The control command calculated by MPC controller presented in Section (4.2.1) is presented in Fig. 6. As can be seen except at the time $t = 15\text{ s}$ where the upper constraint on the control command become active, there is a very good match between the MPC controller and the string stable controller presented in Section (4.1).

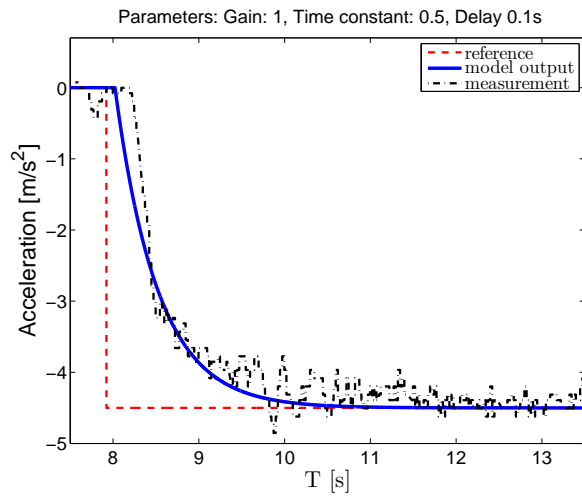
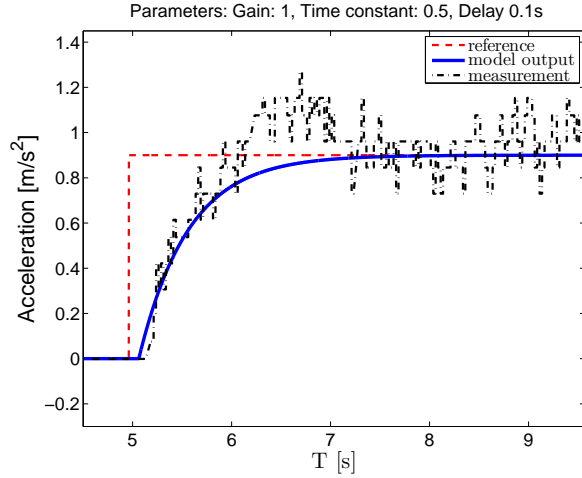


Figure 4: Step response of Volvo S60. The step command, the acceleration measurements and the model output are presented in dashed red, dashed dot black and solid blue, respectively.

6.3. Scenario 2 (string stability)

405 The capability of the proposed approach to guarantee string stability (16) is evaluated both in simulations and experiments. The scenario considered in the simulation is based on the velocity profile introduced in [26].
410 Such scenario mimics a real traffic situation in moderately congested traffic, where acceleration and deceleration are inevitable.

6.3.1. simulations

415 In simulation a reference velocity profile according to Fig. 7 is considered for the leader. In the upper plot of Fig. 7, the velocity profile of the leader and the three followers are shown. The two followers are controlled

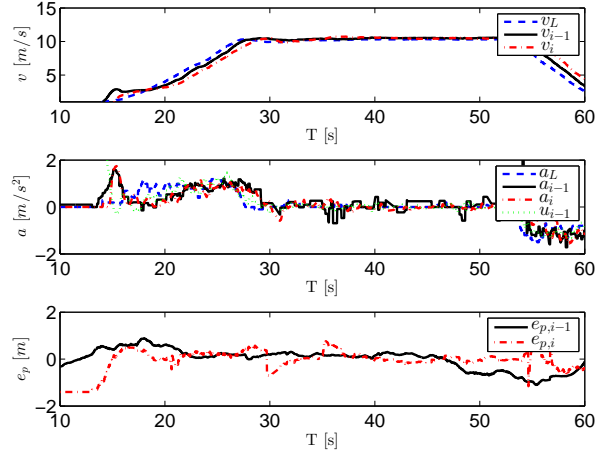


Figure 5: (a) Velocity profile of three vehicles, i.e. lead vehicle in dashed blue, first follower $i - 1$ controlled by an MPC in solid black and the last follower i controlled by MPC dashed dotted red (b) Acceleration profile of leader, first and second follower controlled by an MPC in dashed blue, solid black and dashed dotted red, respectively, dotted green represent the control command of follower $i - 1$ (c) Position error between follower $i - 1$ and leader and between follower i and $i - 1$ in solid black and dashed dotted red, respectively.

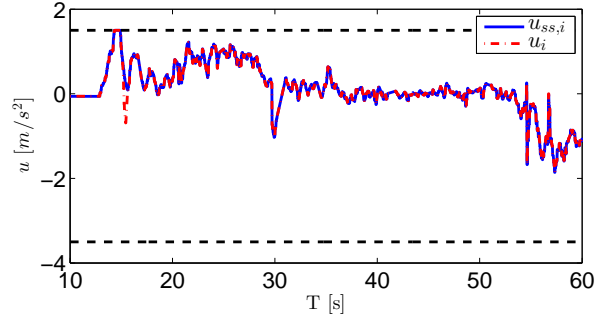


Figure 6: Control signal of MPC controller and String stable controller for the follower i in dashed dotted red and solid blue, respectively. Dashed black represents the constraints.

420 by MPC controllers (22). The results indicate a good velocity tracking without introducing any overshoot. To evaluate string stability, the lead vehicle introduces a speed variation disturbance, Fig. 7. In the middle plot of Fig. 7 the acceleration signals of three followers are depicted. The acceleration is damped out in the platoon which confirms string stability. The position errors of the three mentioned vehicles are also presented to the bottom in Fig. 7. The simulation results also indicate a very satisfactory response in terms of damping the position error in the platoon. The scenario is simulated using both controllers introduced in Section (4), i.e., the string stable $H_{\infty/2}$ and the MPC controller which is tuned according to the control matching tech-

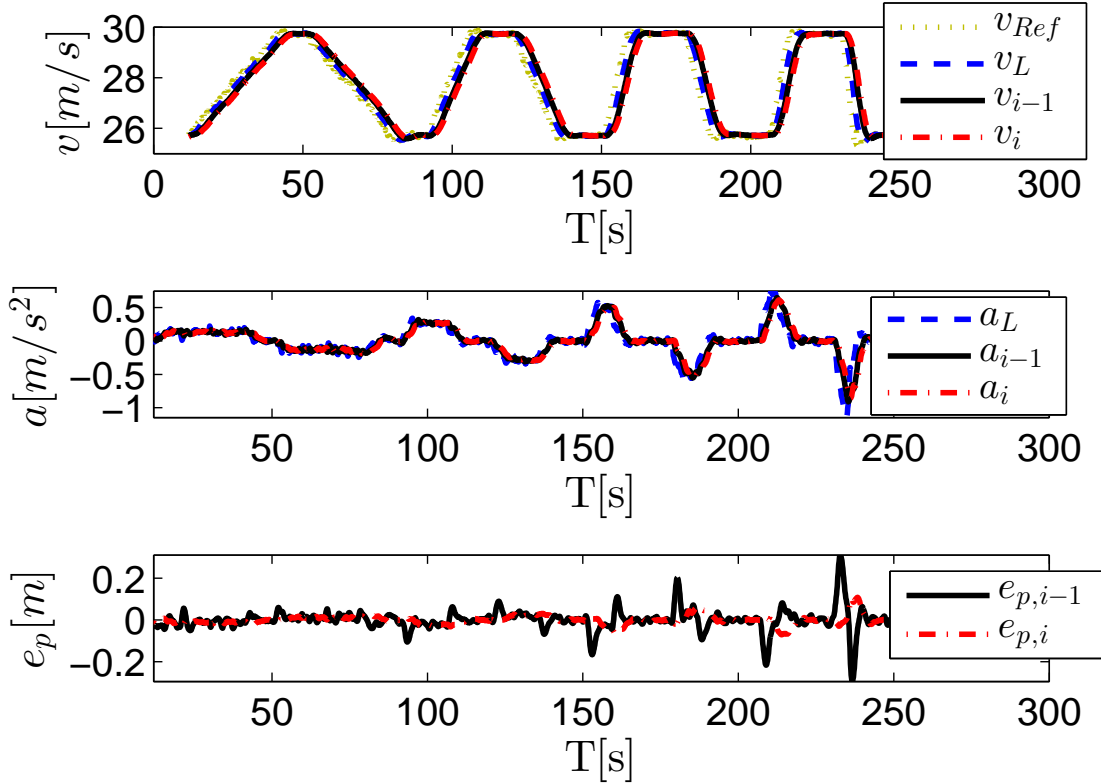


Figure 7: (a) Reference velocity in dashed blue and velocity profile of three vehicles, i.e. lead vehicle in solid black, first follower controlled by an MPC in dashed green and the last follower controlled by MPC dashed dotted red (b) Acceleration profile of leader, first and second follower controlled by an MPC in solid black, dashed green and dashed dotted red, respectively (c) Position error between follower $i - 1$ and leader and between follower i and $i - 1$ in dashed green and dashed dotted red, respectively

nique presented in Section (4.2.1). The control signals of the two controllers for previously described scenarios are presented in Fig. 8. As we expected the control signal of the two controllers are identical, hence the string stability of vehicle platoon with MPC controller is guaranteed as well.

6.3.2. experiments

The effectiveness of proposed method is also evaluated in the experiment with a platoon of three vehicles. We should note that, preferably one would like to repeat similar experiment as the one described in the simulation to evaluate the capability of vehicle platoon in maintaining string stability in practice as well. However, due to the lack of appropriately long test track, the scenario considered here is restricted by the length of available test track.

The lead vehicle is driven manually by a driver and

the speed variation starts after cruising to $40 \frac{km}{h}$, then the driver braked to almost $30 \frac{km}{h}$. This sequence is repeated two times and at the end of the track a slightly harsher brake is applied for approaching to standstill. Followers $i - 1$ and i are controlled by MPC controllers which are developed according to the method presented in Section 4.2.1 to guarantee string stability. In addition to the MPC controller the string stable state feedback/feedforward controller introduced in Section 4 is also implemented on the real time platform. The control action of the string stable controller is logged for the sake of comparison between the two controllers. The results from the experiments are presented in Fig. 9. The measured velocity of the lead vehicle, follower $i - 1$ and i are presented to the top. The acceleration signals are shown in the middle. As can be seen the three vehicles platoon shows a string stable behavior in terms of attenuation of the acceleration signals according to

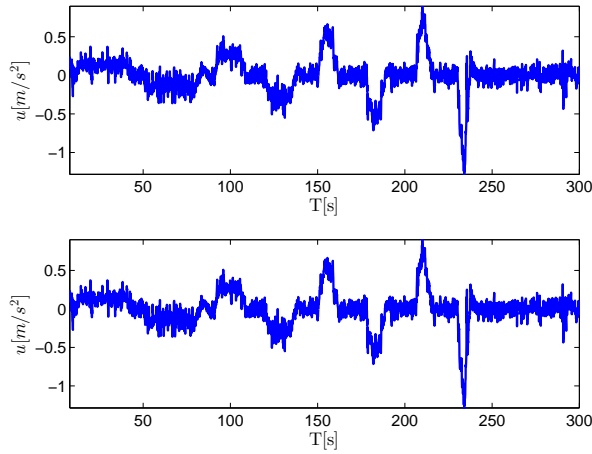


Figure 8: (a) Control command by MPC controller (u_i) (b) control command by $H_{\infty/2}$ ($u_{ss,i}$)

(16). The position errors between vehicles are depicted to the bottom. As can be seen the position error is almost kept below $1m$ during the maneuver. The control signals of MPC controller together with the the control signal generated by the string stable $H_{\infty/2}$ controller are presented in Fig. 10 with dashed dotted red, and solid blue, respectively. This figure indicates that as long as the constraints are not active both controllers, i.e., the MPC controller and $H_{\infty/2}$ controller show identical behaviors.

Furthermore, string stability is also verified in the frequency domain. Hence, the magnitude of $\Gamma_i(j\omega)$ in (16) is estimated using a parametric system identification approach. In particular, an ARX model with three poles and two zeros is considered. The choice of such model is motivated by considering the fact that from (18), the transfer function between a_{i-1} and a_i can be modelled as a third order with two zeros. The magnitude of the estimated transfer function together with the magnitude of theoretical transfer function calculated based on (18) are presented in Fig. 11. The results indicate that the consistency between experimental results and theory.

We should note that the MPC controllers are implemented using fast QP solver generated by CVXGEN, [27]. The sampling time of controllers is $t_s = 0.05sec$ and a prediction horizon $N = 4$ is considered.

Remark 4. In this experiment as can be seen from the middle part of Fig. 9, in addition to the acceleration a_{i-1} , the control command, i.e., u_{i-1} is also sent to the follower i . Hence, the commanded control u_{i-1} is used by the controller to calculate the feedforward action. In general this can result in a quicker feedforward action.

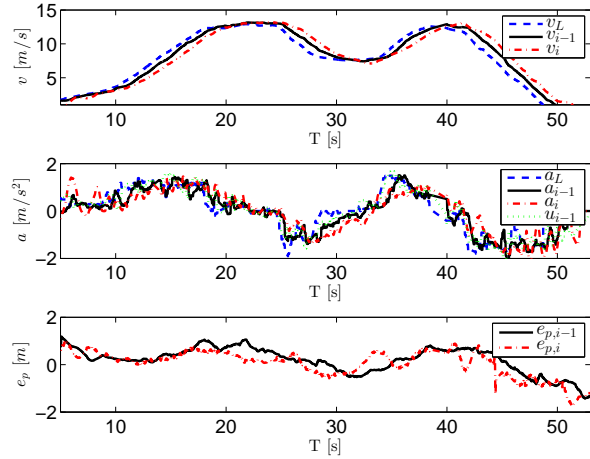


Figure 9: (a) Velocity profile of three vehicles, i.e., lead vehicle in dashed blue, first follower $i - 1$ controlled by an MPC in solid black and the last follower i controlled by MPC dashed dotted red (b) Acceleration profile of leader, first and second follower controlled by an MPC in dashed blue, solid black and dashed dotted red, respectively, dotted green represent the control command of follower $i - 1$ (c) Position error between follower $i - 1$ and leader and between follower i and $i - 1$ in solid black and dashed dotted red, respectively.

6.4. Scenario 3 (constraint satisfaction)

Constraint satisfaction is also evaluated in the experiments with three prototype vehicles. Contrary to the previous scenario where the constraints never became active. In this case, the capability of MPC controllers in handling the constraint is evaluated. In particular, two cases with a platoon of two and three vehicles are considered. The aim of such experiments is to push the vehicle to the limit that is defined by the constraints introduced in Section (3) and subsequently verify the capability of proposed method in handling such constraints.

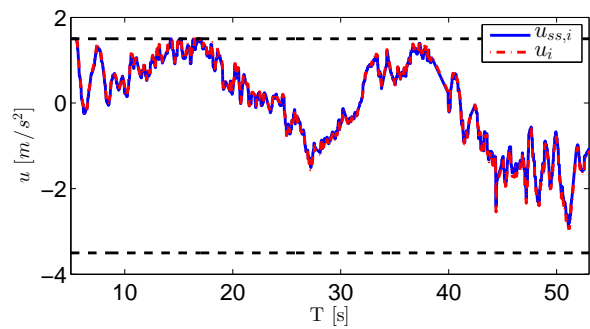


Figure 10: Control signal of MPC controller and String stable controller for the vehicle i in dashed dotted red and solid blue, respectively.

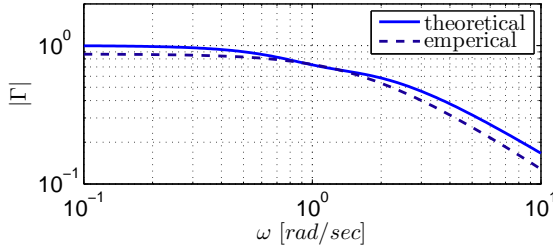


Figure 11: Theoretical and estimated magnitude of frequency response in solid blue and dashed black, respectively.

6.4.1. Two vehicles experiment

510 A velocity profile according to Fig. 12 is considered. Hence, the safety constraint, i.e., constraint on the negative position error is set by the MPC controller. As can be seen from the figure a soft constraint on $e_{p,min} = -0.6m$. The soft constraint is used to avoid infeasibility and a maximum relaxation of $s_{max} = -0.5m$ is used as hard constraint. In this scenario the vehicle is pushed to the constraints. In particular, by looking at Fig. 12, we can see that the constraints become active at $t = 35s$ and $t = 44s$. As we expected when the constraints are active the behaviors of the two controllers are not matched anymore. The control signals of the two controllers are shown in the Fig. 13. When the safety constraint become active, MPC controller commanded the maximum braking capability, at $t = 35sec$, and also a harsh brake at, $t = 44sec$, to avoid violating the constraint. While this is the case for the MPC controller, the string stable $H_{\infty/2}$ controller cannot handle the situation. Commanding the brake by MPC controller results in a deceleration by vehicle which e.g., can be noticed at $t = 35sec$, see Fig. 12. Consequently, the MPC controller managed to keep vehicle within the constraints.

Remark 5. *The scenario considered here is an example to show that with an MPC controller is possible to satisfy constraints arising from our specifications and requirements introduced in Section (3).*

6.4.2. Three vehicles experiment

540 We repeat similar scenario as previous case with a platoon of three vehicles. Similar to previous case soft and hard constraint on the position error and control command are set, respectively. In order to evaluate the effectiveness of proposed method in handling the constraint the second follower is pushed such that the the constraint on the position error become active. This is done by applying a harsh brake at $t = 29s$ by the leader.

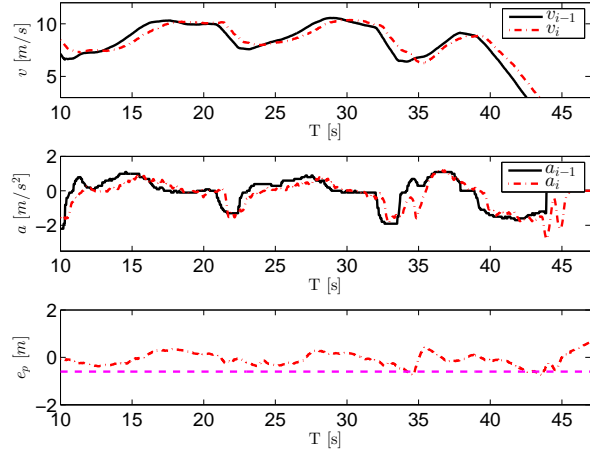


Figure 12: (a) Velocity profile of three vehicles, i.e. lead vehicle in solid green, first follower controlled by an MPC in dashed blue and the second follower controlled by MPC dashed dotted red (b) Position error, i.e. first follower and second follower for MPC and $H_{\infty/2}$ controllers are shown in solid blue and dashed dotted red, respectively. The soft and hard constraints are shown in dashed green and blue, respectively (c) Difference between control signal of MPC controller and String stable controller for the first follower.

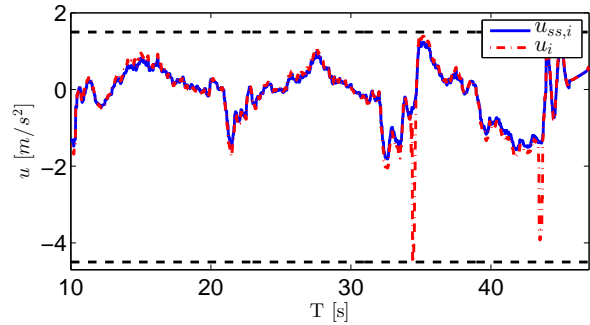


Figure 13: Control signal of MPC controller and String stable controller for the vehicle i in dashed dotted red and solid blue, respectively.

550 Furthermore, we also consider a faulty scenario where the follower i cannot receive the control command of first follower u_{i-1} . This can result in a slower response by follower i . As can be seen from Fig. 14 at $t = 33s$ and $t = 50s$ the position error constraint become active. Hence, as we expected the controller apply a harsh brake to avoid further violation of the constraint, see Fig. 15.

7. Conclusion

555 In this work a control matching approach is used to combine the benefits of frequency domain controller de-

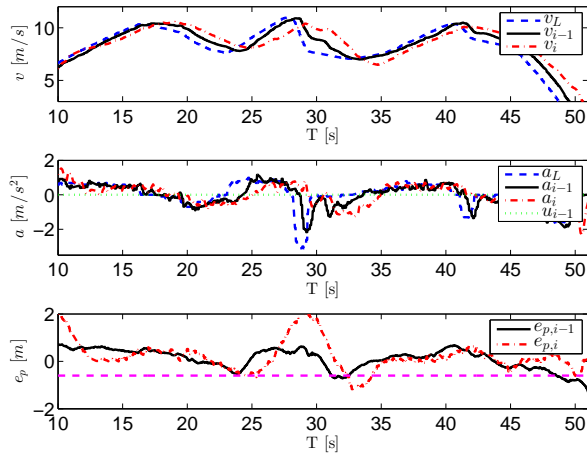


Figure 14: (a) Velocity profile of three vehicles, i.e., lead vehicle in dashed blue, first follower $i - 1$ controlled by an MPC in solid black and the last follower i controlled by MPC dashed dotted red (b) Acceleration profile of leader, first and second follower controlled by an MPC in dashed blue, solid black and dashed dotted red, respectively, dotted green represent the control command of follower $i - 1$ (c) Position error between follower $i - 1$ and leader and between follower i and $i - 1$ in solid black and dashed dotted red, respectively.

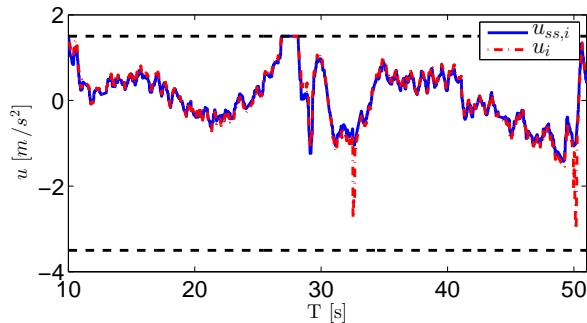


Figure 15: Control signal of MPC controller and String stable controller for the vehicle i in dashed dotted red and solid blue, respectively.

sign, i.e., string stability and benefits of MPC, i.e., constraint satisfaction for a vehicle following application. We showed that by using this approach, the MPC controller can be tuned such that it behaves similar to a string stable $H_{\infty/2}$. Therefore, string stability is guaranteed as long as the constraints are not active. Then it is demonstrated that when the constraints are active the MPC controller is superior. The set that both controllers have identical behavior is the interior of feasible set of MPC controller. This can be characterized by a convex polytope. Proposed method is implemented on the three prototype vehicles and is evaluated in three scenarios, i.e., vehicle following, string stability and constraints

satisfaction. The simulation results together with the experimental results indicate that a single MPC controller can guarantee string stability while respecting the constraints arising from our requirements and specifications.

References

- [1] PATH, California partners for advanced transportation technology (PATH), <http://www.path.berkeley.edu> (1986).
- [2] R. Bishop, A survey of intelligent vehicle applications worldwide, in: Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE, 2000, pp. 25–30.
- [3] A. Alam, A. Gattami, K. Johansson, An experimental study on the fuel reduction potential of heavy duty vehicle platooning, in: Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on, 2010, pp. 306–311. doi:10.1109/ITSC.2010.5625054.
- [4] S. E. Shladover, D. Su, X.-Y. Lu, Impacts of cooperative adaptive cruise control on freeway traffic flow, Journal of the Transportation Research Board 2324.
- [5] B. Van Arem, C. van Driel, R. Visser, The impact of cooperative adaptive cruise control on traffic-flow characteristics, Intelligent Transportation Systems, IEEE Transactions on 7 (4) (2006) 429–436. doi:10.1109/TITS.2006.884615.
- [6] G. Arnaout, S. Bowling, Industrial Engineering and Management.
- [7] R. Rajamani, C. Zhu, Semi-autonomous adaptive cruise control systems, Vehicular Technology, IEEE Transactions on 51 (5) (2002) 1186–1192. doi:10.1109/TVT.2002.800617.
- [8] G. Naus, R. P. Vugts, J. Ploeg, J. de Molengraft, M. Steinbuch, String-stable CACC design and experimental validation: A frequency-domain approach, IEEE Transaction on vehicular technology 59 (9) (2010) 4268–4279.
- [9] F. Bu, H.-S. Tan, J. Huang, Design and field testing of a cooperative adaptive cruise control system, in: American Control Conference (ACC), 2010, 2010, pp. 4616–4621.
- [10] S. Li, K. Li, R. Rajamani, J. Wang, Model predictive multi-objective vehicular adaptive cruise control, Control Systems Technology, IEEE Transactions on 19 (3) (2011) 556–566.
- [11] R. Kianfar, B. Augusto, A. Ebadighajari, U. Hakeem, J. Nilsson, A. Raza, R. S. Tabar, N. V. Irukulapati, C. Englund, P. Falcone, S. Papanastasiou, L. Svensson, H. Wymeersch, Design and experimental validation of a cooperative driving system in the grand cooperative driving challenge, Intelligent Transportation Systems, IEEE Transactions on 13 (3) (2012) 994–1007.
- [12] A. Al Alam, A. Gattami, K. H. Johansson, C. J. Tomlin, Establishing safety for heavy duty vehicle platooning: A game theoretical approach, in: IFAC World Congress, Milan, Italy, 2011.
- [13] R. Kianfar, P. Falcone, J. Fredriksson, Safety verification of automated driving systems, Intelligent Transportation Systems Magazine, IEEE 5 (4) (2013) 73–86. doi:10.1109/MITS.2013.2278405.
- [14] J. Ploeg, N. van de Wouw, H. Nijmeijer, Lp string stability of cascaded systems: Application to vehicle platooning, Control Systems Technology, IEEE Transactions on 22 (2) (2014) 786–793. doi:10.1109/TCST.2013.2258346.
- [15] Grand Cooperative Driving Challenge (GCDC), <http://www.gcdc.net/>.
- [16] J. Eyre, D. Yanakiev, I. Kanellakopoulos, A simplified framework for string stability analysis of automated vehicles, Vehicle System Dynamics 30 (5) (1998) 375–405.

- 630 [17] P. Seiler, A. Pant, K. Hedrick, Disturbance propagation in vehicle strings, *IEEE TRANSACTIONS ON AUTOMATIC CONTROL* 49 (10) (2004) 1835–1841.
- [18] I. Papadimitriou, M. Tomizuka, Lateral control of platoons of vehicles on highways: the autonomous following based approach, *Int. J. Vehicle Design* 36 (1) (2004) 24–37.
- 635 [19] E. Shaw, J. K. Hedrick, Controller design for string stable heterogeneous vehicle strings, 2007 46th IEEE Conference on Decision and Control (2007) 2868–2875.
- [20] W. Dunbar, D. Caveney, Distributed receding horizon control of vehicle platoons: Stability and string stability, *Automatic Control, IEEE Transactions on* 57 (3) (2012) 620–633.
- 640 [21] R. Kianfar, P. Falcone, J. Fredriksson, A distributed model predictive control approach to active steering control of string stable cooperative vehicle platoon, in: *Advances in Automotive Control*, Vol. 7, 2013.
- 645 [22] S. Di Cairano, A. Bemporad, Model predictive control tuning by controller matching, *Automatic Control, IEEE Transactions on* 55 (1) (2010) 185–190.
- [23] R. Rajamani, *Vehicle Dynamics and Control*, Springer, 2005.
- 650 [24] G. J. L. Naus, R. P. A. Vugts, J. Ploeg, M. J. G. van de Molengraft and M. Steinbuch, String-Stable CACC Design and Experimental Validation: A Frequency-Domain Approach, *IEEE Transactions on Vehicular Technology* 59 (9) (2010) 4268–4279.
- 655 [25] J. P. Maschuw, G. C. Keßler, D. Abel, LMI-based control of vehicle platoons for robust longitudinal guidance, in: *17th IFAC World Congress*, Seoul, Korea, 2008.
- [26] V. Milanés, S. Shladover, J. Spring, C. Nowakowski, H. Kawazoe, M. Nakamura, Cooperative adaptive cruise control in real traffic situations, *Intelligent Transportation Systems, IEEE Transactions on* 15 (1) (2014) 296–305. doi:10.1109/TITS.2013.2278494.
- 660 [27] J. Mattingley, S. Boyd, Cvxgen: a code generator for embedded convex optimization, *Optimization and Engineering* 13 (1) (2012) 1–27.
- 665