Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 52-20 (2019) 279–284

Tuning Communication Latency for Distributed Model Predictive Control

Haimin Hu * **Konstantinos Gatsis** * **Manfred Morari** * **George J. Pappas** *

* *Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104 USA, {haiminhu, kgatsis, morari, pappasg}@seas.upenn.edu*

Abstract: This paper presents a distributed MPC scheme for the class of input-coupled linear systems implemented over wireless networks. The approach allows each agent to achieve reduced communication latency by sending less information to their neighbours. Uncertainties incurred by this delayed and incomplete information are handled by a constraint tightening procedure. Simulation examples demonstrate that for distributed systems with chain structure, our tightening method is invariant to the number of agents in the network. Moreover, we show that by a proper tuning of latency, an optimized control performance can be achieved.

Keywords: distributed systems, predictive control, robust control, communication, latency

1. INTRODUCTION

With the emergence of the Internet-of-Things and the next generation wireless communication networks, there is an increased interest in enabling applications where multiple distributed agents perform cooperative control tasks. Examples include vehicle platooning Dolk et al. (2017); Firoozi et al. (2018); Hu et al. (2018); Vukadinovic et al. (2018), coordination of mobile robots Farina et al. (2015) and UAV formation control Chen et al. (2015). As such applications are envisioned for larger networks or involving faster dynamics, the communication latency experienced by the agents may jeopardize control performance and give rise to safety issues in closed-loop systems. Understanding and mitigating latency is arising as an important challenge in those domains Gatsis et al. (2018); Jiang et al. (2018); Eisen et al. (2019); Maity et al. (2019).

Distributed model predictive control (DMPC) has been a powerful technique for multi-agent control systems, since it explicitly handles state and input constraints as well as the coupling between subsystems, see the survey paper Scattolini (2009). In terms of computation and communication, DMPC methods can be classified into two categories: non-iterative and iterative. Although iterative schemes such as Conte et al. (2012) can achieve control performance close to centralized MPC, it may require up to hundred rounds of communications among agents to reach convergence, which is not suitable for delay-sensitive applications. On the contrary, in non-iterative schemes agents communicate predicted trajectories only once per each sampling period. This on one hand greatly reduces the amount of communication load, but on the other hand introduces uncertainties in the transmitted information, since agents do not necessarily execute exactly as communicated. To guarantee constraint satisfaction, robust MPC (RMPC) is used which treats the uncertain intention of the neighbours as disturbances, see for example Scattolini (2009). Such scheme is also vulnerable to latency since the uncertainty in communicated information grows as the amount of delay increases. However, non-iterative DMPC methods addressing this issue have not been well-developed, with few exceptions including Hahn et al. (2018), where latency is predicted and used in local MPC computation to optimize the control performance, but dynamic coupling between subsystems is not considered.

In this paper, we develop a non-iterative DMPC scheme for distributed linear systems with input coupling. We propose for the first time a mechanism of reducing communication latency by sending shorter predicted trajectories in the context of DMPC. This is in contrast to standard MPC developed for networked control systems, e.g. Quevedo and Nesic (2011), where agents always communicate the entire trajectory. Stability and recursive feasibility are guaranteed using robust MPC methods with a tailored constraint tightening procedure, which takes latency as a parameter. The proposed DMPC scheme allows us to explore the tradeoff between latency and uncertainty of communicated information across agents in order to obtain an optimized control performance without sacrificing safety.

The reminder of the paper is structured as follows. Section 2 reviews the non-iterative distributed MPC scheme. Section 3 explains how latency is tuned for DMPC. In Section 4 a constraint tightening procedure is presented and property of the controller is examined. Section 5 shows the simulation results. Section 6 concludes the paper.

Notation: The set of integers ranging from a to b is denoted by $\mathcal{I}_{a:b}$. The concatenation of vectors $x_i \in \mathbb{R}^{n_i}$ is defined by $\mathrm{col}_{i\in\mathcal{I}_{a:b}} x_i = \mathrm{col}(x_a, \ldots, x_b) = \left[x_a^\top, \ldots, x_b^\top\right]^\top$. The Minkowski sum is defined by $\mathbb{X} \oplus \mathbb{Y} = \{x+y | x \in \mathbb{X}, y \in \mathbb{Y}\}$. The Pontryagin difference is $\mathbb{X} \ominus \mathbb{Y} = \{z | z + \mathbb{Y} \subseteq \mathbb{X}\}$.

⋆ This work is partially supported by NSF CPS-1837253 and by the Intel Science and Technology Center for Wireless Autonomous Systems (ISTC-WAS).

10.1016/j.ifacol.2019.12.194

2. PROBLEM FORMULATION

We consider a discrete-time linear dynamical system

$$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t \tag{1}$$

with the state vector $\mathbf{x}_t \in \mathbb{R}^n$ and the input vector $\mathbf{u}_t \in \mathbb{R}^m$. We refer to (1) as the global system, with $\mathbf{A} \in \mathbb{R}^{n\times n}$ and $\mathbf{B} \in \mathbb{R}^{m\times n}$ as the system and input matrix, respectively. Consider the partition of (1) into M *input-coupled* subsystems with the following dynamics

$$\Sigma_i: \; x_{t+1}^{[i]} = A_i x_t^{[i]} + B_{ii} u_t^{[i]} + \sum\nolimits_{j\in\mathcal{N}_i} B_{ij} u_t^{[j]} \tag{2}$$

where $x_t^{[i]} \in \mathbb{R}^{n_i}$ and $u_t^{[i]} \in \mathbb{R}^{m_i}$ are the state and input vector of subsystem Σ_i, such that $\mathbf{x}_t = \mathrm{col}_{i\in\mathcal{I}_{1:M}} x_t^{[i]}$, $\sum_{i=1}^M n_i = n$ and $\mathbf{u}_t = \mathrm{col}_{i\in\mathcal{I}_{1:M}} u_t^{[i]}$, $\sum_{i=1}^M m_i = m$. The global system matrix $\mathbf{A} = \mathrm{diag}_{i\in\mathcal{I}_{1:M}}[A_i]$, and $B_{ij} \in \mathbb{R}^{n_i\times m_j}$ is the corresponding block in the global input matrix $\mathbf{B}$. Two sets defining the neighbours are introduced. The predecessor set $\mathcal{N}_i = \{j \in \mathcal{I}_{1:M}\backslash\{i\} | B_{ij} \neq 0\}$ contains the indices of neighbouring subsystems of Σ_i, whose control action will affect Σ_i. Likewise the follower set is defined as $\bar{\mathcal{N}}_i = \{j \in \mathcal{I}_{1:M}\backslash\{i\} | B_{ji} \neq 0\}$.

Assumption 2.1. The agent Σ_i only receives information from neighbours in its predecessor set $\mathcal{N}_i$, and sends information to the follower set $\bar{\mathcal{N}}_i$. It is further assumed that the information flow between subsystems is *acyclic*.

Remark 2.1. The assumption on acyclic network is due to the unidirectional information flow in Algorithm 2. An improved version of this algorithm is under our current investigation to allow for bidirectional communication.

Subsystem (2) is subject to state and input constraints

$$x_t^{[i]} \in \mathbb{X}^{[i]}, \; u_t^{[i]} \in \mathbb{U}^{[i]}, \quad \forall i \in \mathcal{I}_{1:M} \tag{3}$$

where $\mathbb{X}^{[i]}$ and $\mathbb{U}^{[i]}$ are polytopic sets containing the origin. This leads to the constraints $\mathbf{x}_t \in \mathbb{X} = \mathbb{X}^{[1]} \times \cdots \times \mathbb{X}^{[M]}$ and $\mathbf{u}_t \in \mathbb{U} = \mathbb{U}^{[1]} \times \cdots \times \mathbb{U}^{[M]}$ on the global system (1).

2.1 Running example

We use heavy duty vehicle platooning as our running example throughout the paper. Consider the constant-spacing model from Dold and Stursberg (2009). The leading vehicle Σ_1 in the platoon has a single integrator dynamics $v_{t+1}^{[1]} = v_t^{[1]} + \delta_t u_t^{[1]}$, whose velocity is controlled by a human driver or a standalone controller. Input u is acceleration. δ_t is the sampling time. States of the following vehicles Σ_i are $x^{[i]} = \begin{bmatrix} e_d^{[i]} & e_v^{[i]} \end{bmatrix}^\top$, the error in distance and velocity with respect to the predecessor Σ_j. According to (2), the model of vehicle Σ_i is given by

$$x_{t+1}^{[i]} = \begin{bmatrix} 1 & \delta_t \\ 0 & 1 \end{bmatrix} x_t^{[i]} + \begin{bmatrix} -{\delta_t}^2/2 \\ -\delta_t \end{bmatrix} u_t^{[i]} + \begin{bmatrix} \delta_t^2/2 \\ \delta_t \end{bmatrix} u_t^{[j]} \tag{4}$$

The desired inter-vehicle distance d_0 and cruising speed v_0 are pre-specified parameters. The control goal is to bring the states of all subsystems to the origin. It is assumed that each vehicle can measure their current states.

Now the global system (1) has a *chained* network topology, with the dynamics of each subsystem defined as

$$\Sigma_i: \; x_{t+1}^{[i]} = A_i x_t^{[i]} + B_{ii} u_t^{[i]} + B_{ij} u_t^{[j]} \tag{5}$$

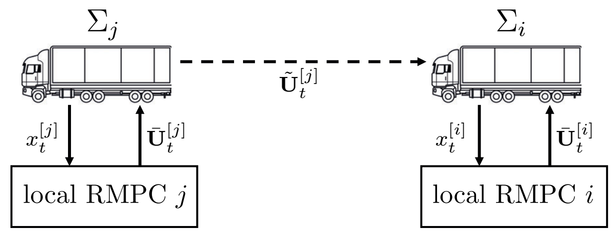

Fig. 1. An example of non-iterative DMPC applied to heavy duty vehicle platooning.

which is a special case of (2). The predecessor set becomes

$$\mathcal{N}_i = \begin{cases} \emptyset & i = 1 \\ \{j\} := \{i-1\} & i \geq 2 \end{cases} \tag{6}$$

similarly the follower set $\bar{\mathcal{N}}_i$ is now a singleton as well. Ways to extend our approach to more general networks are discussed in Remark 4.1.

2.2 Non-iterative DMPC

To enable distributed control of the global system (1), we propose to use the non-iterative DMPC scheme, as illustrated in Figure 1. In this framework, each agent first solves in parallel a local MPC problem, and then communicates the solution, i.e. a sequence of predicted inputs, to the neighbours. In the presence of latency, agents are uncertain about future intentions of the neighbours since their currently planned trajectories may differ from the ones that have been communicated previously.

In order to guarantee constraint satisfaction despite the worst-case uncertainty, we adopt the RMPC technique proposed in Chisci et al. (2001). The local RMPC problem of subsystem Σ_i to be solved at time t is given by

$$\min_{\bar{\mathbf{X}}_t^{[i]}, \bar{\mathbf{V}}_t^{[i]}} \sum_{k=t}^{t+N-1} (\bar{v}_k^{[i]})^\top L_i \bar{v}_k^{[i]} \tag{7a}$$

$$\text{s.t.} \quad \bar{x}_t^{[i]} = x_t^{[i]} \tag{7b}$$

$$\bar{x}_{k+1}^{[i]} = \Phi_i \bar{x}_k^{[i]} + B_{ii}\bar{v}_k^{[i]} + B_{ij}\tilde{u}_k^{[j]} \tag{7c}$$

$$\bar{x}_{k+1}^{[i]} \in \bar{\mathbb{X}}_{k+1}^{[i]}, \; \bar{u}_{k+1}^{[i]} \in \bar{\mathbb{U}}_{k+1}^{[i]} \tag{7d}$$

$$g(\bar{\mathbf{V}}_t^{[i]}) \leq 0, \quad \forall k \in \mathcal{I}_{t:t+N-1} \tag{7e}$$

where $\bar{x}_k^{[i]}$ and $\bar{v}_k^{[i]}$ are decision variables, in compact form $\bar{\mathbf{X}}_t^{[i]} = \mathrm{col}_{k\in\mathcal{I}_{t:t+N}} \bar{x}_k^{[i]}$, similarly for $\bar{\mathbf{V}}_t^{[i]}$. $L_i \in \mathbb{S}_+^{m_i\times m_i}$ is a positive definite weighting matrix. In the dynamics constraint (7c), prediction of $\bar{x}_k^{[i]}$ relies on future inputs $\tilde{u}_k^{[j]}$ computed and transmitted by neighbours Σ_j with $j \in \mathcal{N}_i$. We consider the following parameterization of inputs

$$\bar{u}_k^{[i]} = \bar{v}_k^{[i]} + K_i \bar{x}_k^{[i]} \tag{8}$$

where K_i is a fixed feedback gain. Consequentially, the local dynamics (2) can be rewritten as (7c) where the closed-loop system matrix is defined as

$$\Phi_i = A_i + B_{ii}K_i \tag{9}$$

In case of nominal feedback, i.e. $\bar{v}_k^{[i]} = 0, \forall i \in \mathcal{I}_{1:M}$, the global system (1) can be expressed as $\mathbf{x}_{t+1} = \mathbf{\Phi}\mathbf{x}_t$ where the matrix $\mathbf{\Phi}$ is structured with $A_i + B_{ii}K_i$ on the diagonal and $B_{ij}K_j$ as the lower off-diagonal blocks.

Assumption 2.2. The matrix $\mathbf{\Phi}$ is Schur.

To ensure robust constraint satisfaction, we define in (7d) a series of tightened constraint sets

$$\begin{aligned} \bar{\mathbb{X}}_k^{[i]} &= \mathbb{X}^{[i]} \ominus \mathbb{W}_k^{[i]}, \quad \bar{\mathbb{X}}_N^{[i]} = \mathbb{X}_F^{[i]} \ominus \mathbb{W}_N^{[i]} \\ \bar{\mathbb{U}}_k^{[i]} &= \mathbb{U}^{[i]} \ominus K_i \mathbb{W}_k^{[i]} \end{aligned} \tag{10}$$

for all $k \in \mathcal{I}_{1:N-1}$. $\mathbb{X}_F^{[i]}$ is the terminal set which can be chosen as a positive invariant set of (5). The time-varying disturbance sets $\mathbb{W}_k^{[i]}$ bounds the uncertainty in received inputs $\tilde{u}_{t+k}^{[j]}$. The convex constraint (7e) bounds the differences between the transmitted trajectories at consecutive times. One of the main contributions of this paper is on how to select those bounds in a manner that accounts for uncertainties stemming from communication latency and transmitted information among agents. These aspects will be discussed in detail in Section 4.

Optimization problem (7) returns for each subsystem Σ_i a sequence of inputs $\bar{\mathbf{U}}_t^{[i]} = \mathrm{col}(\bar{u}_{t|t}^{[i]}, \dots, \bar{u}_{t+N-1|t}^{[i]}) \in \mathbb{R}^N$, where we denote the solution as $\bar{(\cdot)}_{t_1|t_2}$, with t_1 as the internal time step of MPC, and t_2 as the time when (7) is solved. Only the first input is applied, i.e. $u_t^{[i]} = \bar{u}_t^{[i]}$.

3. TUNING LATENCY FOR DMPC

In this paper at each time step each agent sends a sequence of predicted control inputs to its follower. Our aim is to capture the effect of this communication load on the communication latency. More specifically we adopt a model where the latency d of communication is a function of the length of predicted control inputs that need to be communicated, $d = d(N_s)$ – see also Remark 3.1 for a justification. For simplicity we consider this latency model to be linear (see also Figure 4 in simulations). This relationship enables us to adapt the length of transmitted information as a mechanism to reduce latency.

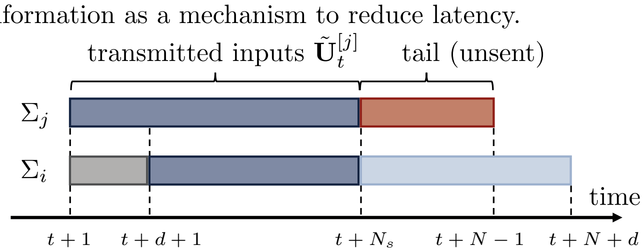

Fig. 2. At time t agent Σ_j transmits the sequence $\tilde{\mathbf{U}}_t^{[j]}$. The neighbour Σ_i receives the copy at time $t+d+1$, and reconstructs the remaining $N - N_s + d$ inputs, as shown in light blue on the bottom.

As shown in Figure 2, now the sequence $\bar{\mathbf{U}}_t^{[j]}$ is divided into two parts based on the choice of transmitted packet length N_s. The first part contains the future inputs of Σ_j transmitted to its neighbour agent Σ_i, which is defined as

$$\tilde{\mathbf{U}}_t^{[j]} = \mathrm{col}(\tilde{u}_{t+1|t}^{[j]}, \dots, \tilde{u}_{t+N_s|t}^{[j]}) \in \mathbb{R}^{N_s}.$$

where $\tilde{u}_{t+k|t}^{[j]} = \bar{u}_{t+k|t}^{[j]}$ with an abuse of notation. Due to causality, agents would receive future plans from their neighbours at least one time step after the information has been sent. Therefore, we do not send the first input $\bar{u}_{t|t}^{[j]}$.

For the second part we introduce the notion of *tail indices*

$$\mathcal{I}_T = \mathcal{I}_{N_s+1:N-1}$$

and correspondingly the tail part of $\bar{\mathbf{U}}_t^{[j]}$ is

$$\mathrm{col}(\bar{u}_{t+N_s+1|t}^{[j]}, \dots, \bar{u}_{t+N-1|t}^{[j]}) \in \mathbb{R}^{N-N_s-1}$$

which is not communicated by Σ_j and of course, unknown to Σ_i. In the next section we will develop methods to bound this uncertainty. Subsystem Σ_i receives the sequence $\tilde{\mathbf{U}}_t^{[j]}$ at time $t+d+1$. Recall in constraint (7c) that an input sequence of Σ_j with length N is required for MPC computation of Σ_i. But the number of inputs available to Σ_i is now only $N_s - d$, because the first d inputs (grey bar in Figure 2) in $\tilde{\mathbf{U}}_t^{[j]}$ are useless for the MPC problem of Σ_i at current time $t+d+1$. To resolve this issue, we use the following input sequence of Σ_j in the local RMPC computation of Σ_i

$$\hat{\mathbf{U}}_{t+d+1}^{[j]} = \mathrm{col}(\tilde{u}_{t+d+1|t}^{[j]}, \dots, \tilde{u}_{t+N_s|t}^{[j]}, \tilde{u}_{t+N_s+1}^{[j]}, \dots, \tilde{u}_{t+N+d}^{[j]})$$

with the first $N_s - d$ elements extracted out from $\tilde{\mathbf{U}}_t^{[j]}$, and the remaining $N - N_s + d$ elements being reconstructed by setting $\bar{v}_k^{[j]} = 0$ and using the dynamics equation with the feedback gain K_j of the neighbour, i.e.

$$\tilde{u}_k^{[j]} = K_j \tilde{x}_k^{[j]}, \ \tilde{x}_{k+1}^{[j]} = \Phi_j \tilde{x}_k^{[j]}, \quad \forall k \in \mathcal{I}_{t+N_s+1:t+N+d} \tag{11}$$

It is easy to verify that $\hat{\mathbf{U}}_{t+d+1}^{[j]} \in \mathbb{R}^N$.

Remark 3.1. The length of predicted inputs to be sent among agents directly affects the length of the packets to be transmitted, and subsequently the packet length affects latency. This is justified in multiple scenarios. To achieve reliable communication under adverse channel conditions such as deep fades, packets need to be retransmitted multiple times until successful reception. In this case, longer packets directly imply longer delays. Additionally, in cases where multiple agents need to communicate over a shared multiple access channel, there is a latency associated with the multiple access procedure and this is again dependent on the packet length. For example, in a V2X application where multiple vehicles exchange information with each other or with the infrastructure based on DSRC protocol, the employed multiple access scheme is CSMA (Li (2010)) which means that each agent needs to wait for a random time until it finds the channel clear from other packet transmissions. As a limitation, we point out that our approach of controlling latency by reducing the packet length may have a limited impact in practical scenarios where the payload is already small, or under ideal channel conditions, or in communication networks with small number of users. Managing latency in next generation wireless networks is an ongoing research topic (Jiang et al. (2018)).

Based on the non-iterative DMPC scheme introduced in Section 2 and the latency tuning procedure discussed in this section, we provide an online control loop for each subsystem Σ_i in Algorithm 1.

Algorithm 1 Online control of subsystem Σ_i

Input: Current state $x_t^{[i]}$, delayed sequence $\tilde{\mathbf{U}}_{t-d-1}^{[j]}$

1: Reconstruct the input sequence $\hat{\mathbf{U}}_t^{[j]}$ via (11)
2: Solve the local RMPC problem (7) and obtain $\bar{\mathbf{U}}_t^{[i]}$
3: Apply the first control input $\bar{u}_t^{[i]}$
4: Truncate $\bar{\mathbf{U}}_t^{[i]}$ to obtain $\tilde{\mathbf{U}}_t^{[i]}$ and transmit to Σ_{i+1}

4. PROPOSED DMPC SCHEME

In this section, we describe how distributed MPC may incorporate our strategy of tuning latency as proposed in Section 3. The state and input constraints in the MPC formulation (7) now depend on the choices of latency parameters d and N_s. We point out that there are two sources of uncertainties in the MPC problem:

- Latency: Mismatch between $\tilde{u}^{[j]}_{t+k|t-d-1}$, the transmitted inputs delayed by $d+1$ time steps, and $\bar{u}^{[j]}_{t+k|t}$, the inputs currently predicted by Σ_j
- Reconstructed inputs: $\tilde{u}^{[j]}_{t+k}, \forall k \in \mathcal{I}_{N_s+1:N+d}$, the part of inputs of Σ_j that is currently unknown to Σ_i, as illustrated by the light blue bar in Figure 2

Those two kinds of uncertainties are at odds with each other: one can either send more information at the cost of larger delay, or achieve lower latency by sending less information to the neighbour. In fact, latency parameters d and N_s are the 'handle' balancing one from the other.

Note that both uncertainties would lead to error in predicting the states $x^{[i]}_{t+k}$ for Σ_i. To see this, first recall the prediction model (7c) in local RMPC of Σ_i

$$\bar{x}^{[i]}_{t+k+1} = \Phi_i x^{[i]}_{t+k} + B_{ii}\bar{v}^{[i]}_{t+k} + B_{ij}\tilde{u}^{[j]}_{t+k} \tag{12}$$

and consider the 'omniscient' prediction model which knows the input sequence of Σ_j without any delay

$$x^{[i]}_{t+k+1} = \Phi_i x^{[i]}_{t+k} + B_{ii}\bar{v}^{[i]}_{t+k} + B_{ij}\bar{u}^{[j]}_{t+k|t+k} \tag{13}$$

We consider the difference between $x^{[i]}_{t+k}$ and $\bar{x}^{[i]}_{t+k}$ as artificial disturbances. By subtracting (12) from (13), the initial disturbance $w^{[i]}_{t+1}$ at time $t+1$ is obtained as

$$w^{[i]}_{t+1} = x^{[i]}_{t+1} - \bar{x}^{[i]}_{t+1} = B_{ij}\left(u^{[j]}_{t|t} - \tilde{u}^{[j]}_{t}\right) \in \mathbb{W}^{[i]}_1 \tag{14}$$

The subsequent disturbance for all $k \in \mathcal{I}_{1:N-1}$ is

$$w^{[i]}_{t+k+1} = \Phi_i w^{[i]}_{t+k} + B_{ij}\left(u^{[j]}_{t+k|t} - \tilde{u}^{[j]}_{t+k}\right) \in \mathbb{W}^{[i]}_k \tag{15}$$

with $\mathbb{W}^{[i]}_0 = \emptyset$ due to the constraint (7b) on initial states. To this end, we have modeled the uncertainty caused by communication as time-varying persistent disturbances. Its support over time, as we will show, can be bounded within polytopic sets $\mathbb{W}^{[i]}_k$. This allows us to use the RMPC framework proposed in Chisci et al. (2001).

4.1 Computation of the disturbance sets

Based on latency parameters d and N_s, we now present a complete version of the MPC problem (7). The constraint $g(\bar{\mathbf{V}}^{[i]}_t) \leq 0$ in (7e) is defined as

$$\begin{aligned} \bar{v}^{[i]}_{t+k|t} - \bar{v}^{[i]}_{t+k|t-1} &\in \mathbb{V}^{[i]}, \quad \forall k \in \mathcal{I}_{0:N-1} \setminus \mathcal{I}_T \\ \bar{v}^{[i]}_{t+k|t} - 0 &\in \mathbb{T}^{[i]}, \quad \forall k \in \mathcal{I}_T \end{aligned} \tag{16}$$

where $\mathbb{V}^{[i]}, \mathbb{T}^{[i]} \subseteq \mathbb{R}^{m_i}$ are pre-defined polytopic sets. The intuition behind (16) is that the deviation of current plan $\bar{v}^{[i]}_{t+k|t}$ from the last plan $\bar{v}^{[i]}_{t+k|t-1}$ is bounded within $\mathbb{V}^{[i]}$. Furthermore, if $\bar{v}^{[i]}_{t+k}$ is in the tail, i.e. $k \in \mathcal{I}_T$, the variable is restricted to be contained in $\mathbb{T}^{[i]}$, a neighbourhood around the origin. The sets $\mathbb{V}^{[i]}$ and $\mathbb{T}^{[i]}$ are the key for deriving the disturbance sets $\mathbb{W}^{[i]}_k$. They on one hand restrict the movement and shrink the feasible set of agent Σ_i, but on the other hand considerably reduce the uncertainty in transmitted inputs $\tilde{u}^{[i]}_t$ for the neighbour Σ_{i+1}, making the disturbance bound less conservative. For each subsystem $\Sigma_i, \forall i \in \mathcal{I}_{2:M}$, the disturbance sets $\mathbb{W}^{[i]}_k, \forall k \in \mathcal{I}_{1:N}$ are defined by the following recursion

$$\mathbb{W}^{[i]}_{k+1} = \Phi_i \mathbb{W}^{[i]}_k \oplus B_{ij}\hat{\mathbb{U}}^{[j]}_k, \quad \forall k = \mathcal{I}_{1:N-1} \tag{17a}$$

$$\mathbb{W}^{[i]}_1 = B_{ij}\left[(d+1)\mathbb{V}^{[j]} \oplus K_j \hat{\mathbb{X}}^{[j]}_0\right] \tag{17b}$$

where

$$\hat{\mathbb{U}}^{[j]}_k = \hat{\mathbb{V}}^{[j]}_k \oplus K_j \hat{\mathbb{X}}^{[j]}_k \tag{18}$$

Note that for the leading agent Σ_1 we have $\mathbb{W}^{[1]}_k = \emptyset$ since $\mathcal{N}_1 = \emptyset$. The recursive definition of $\mathbb{W}^{[i]}_k$ in (17) comes directly from the disturbance expression (14) and (15). Uncertainty in the transmitted inputs makes up $\mathbb{W}^{[i]}_k$ over time, as suggested by (17a). Recall in (8) that the input can be separated into two parts, i.e. $\bar{u}^{[j]}_k = \bar{v}^{[j]}_k + K_j\bar{x}^{[j]}_k$, which we can give bounds respectively. The first part bounds the uncertainty in $\bar{v}^{[j]}_k$, defined as

$$\hat{\mathbb{V}}^{[j]}_k = \begin{cases} (d+1)\mathbb{V}^{[j]}, & \forall k \in \mathcal{I}_{1:N_s-1} \\ \mathbb{T}^{[j]} \oplus m_k \mathbb{V}^{[j]}, & \forall k \in \mathcal{I}_{N_s:N_s+d} \\ \mathbb{T}^{[j]}, & \forall k \in \mathcal{I}_T \end{cases} \tag{19}$$

where $m_k = N_s + d - k + 1$. Due to (16), the worst-case deviation between the transmitted inputs and the one currently predicted by neighbour Σ_j is bounded within a multiple of $\mathbb{V}^{[j]}$, which grows linearly with latency d. The second part gives uncertainty bound on $\bar{x}^{[j]}_k$, i.e. the difference between the states of neighbour Σ_j encoded in its transmitted input, and the states currently predicted by it. The difference is bounded by

$$\hat{\mathbb{X}}^{[j]}_{k+1} = \Phi_j \hat{\mathbb{X}}^{[j]}_k \oplus B_{jj}\hat{\mathbb{V}}^{[j]}_k \oplus B_{jj-1}\hat{\mathbb{U}}^{[j-1]}_k \tag{20a}$$

$$\hat{\mathbb{X}}^{[j]}_0 = \bigoplus_{k=1}^{d} \Phi_j^{k-1}\left[B_{jj}(d+1)\mathbb{V}^{[j]} \oplus B_{jj-1}\hat{\mathbb{U}}^{[j-1]}_{d-k}\right] \tag{20b}$$

for all $k \in \mathcal{I}_{0:N-1}$. For the second agent Σ_2 in the chain we have $\hat{\mathbb{U}}^{[j-1]}_k = \emptyset$. Here the set $\hat{\mathbb{U}}^{[j-1]}_k$ captures the coupling effect from the neighbour's neighbour, namely agent Σ_{j-1}. A summary of the offline design procedure of the DMPC controller is given in Algorithm 2.

Remark 4.1. Algorithm 2 assumes the global system (1) is in chain structure. Nevertheless it can be easily generalized to the case of multiple neighbours by taking Minkowski sum of the sets in Step 2 over all predecessors.

Algorithm 2 Offline design of DMPC

Input: Latency parameters d and N_s, system model (1) and sets $\mathbb{V}^{[i]}$, $\mathbb{T}^{[i]}$

```
1: for subsystem index i ← 2 to M do
2:     Compute sets W_k^[i] via (17), (18), (19) and (20)
3:     Obtain tightened sets X̄_k^[i] and Ū_k^[i] via (10)
4:     Pass the sets Û_k^[j] computed in (18) to Σ_{i+1}
5: end for
```

4.2 Feasibility and stability

This section states the main properties of the proposed DMPC controller.

Lemma 4.1. Consider the following statements for all agents in the global system (1). For each agent Σ_i let $(\bar{\mathbf{X}}_t^{[i]}, \bar{\mathbf{V}}_t^{[i]})$ be the solution to (7) at time t, to which the inputs are the current state $x_t^{[i]} \in \mathbb{R}^{n_i}$ and sequence $\hat{\mathbf{U}}_t^{[j]} \in \mathbb{R}^{N-1}$ reconstructed from the transmitted inputs $\tilde{\mathbf{U}}_{t-d-1}^{[i]} \in \mathbb{R}^{N_s}$ of neighbour Σ_j with latency d. Consider state evolution $x_{t+k+1}^{[i]} = A_i x_{t+k}^{[i]} + B_{ii} u_{t+k}^{[i]} + B_{ij} u_{t+k}^{[j]}$ with input $u_{t+k}^{[i]} = \bar{v}_{t+k|t}^{[i]} + K_i x_{t+k}^{[i]}$, then it holds that

$$x_{t+k}^{[i]} - \bar{x}_{t+k|t}^{[i]} \in \mathbb{W}_k^{[i]}, \qquad \forall k \in \mathcal{I}_{1:N} \tag{21a}$$

$$u_{t+k}^{[i]} - \bar{u}_{t+k|t}^{[i]} \in K_i \mathbb{W}_k^{[i]}, \qquad \forall k \in \mathcal{I}_{1:N-1} \tag{21b}$$

Proof. See Appendix A.

Assumption 4.1. The accumulated deviation on $\bar{v}_{t+k|t}^{[i]}$ is bounded by $\bar{v}_{t+k|t}^{[i]} - \bar{v}_{t'+k|t'}^{[i]} \in (d+1)\mathbb{V}^{[i]}$ for all $k \in \mathcal{I}_{0:N-1}$. $t' \leq t$ is the last time when $\bar{x}_{t'+k+1|t'}^{[i+1]} \oplus \mathbb{W}_{k+1}^{[i+1]} \subseteq \bar{\mathbb{X}}_{k+1}^{[i+1]}$.

Theorem 4.1. (Recursive feasibility). Let Assumption 4.1 hold. For each agent $\Sigma_i, \forall i \in \mathcal{I}_{1:M}$ under control law (8), if (7) is feasible for $x_t^{[i]}$, then (7) is feasible for $x_{t+1}^{[i]}$.

Proof. Lemma 4.1 and Lemma 7 in Chisci et al. (2001) imply that the solution to (7) at time t shifted by one time step is also feasible at $t+1$, if $\hat{\mathbf{U}}_t^{[j]}$ is considered in the prediction model (7c). Moreover, Assumption 4.1 prevents infeasibility in (7d) when the initial state is reset to the current state (see (7b)) and (7c) is updated by $\hat{\mathbf{U}}_{t+1}^{[j]}$.

Remark 4.2. (Stability). For each agent Σ_i, $\forall i \in \mathcal{I}_{1:M}$ under MPC control law (8) starting from an initial state $x_0^{[i]}$ for which (7) is feasible, the state $x_t^{[i]}$ is convergent to the origin. The idea of proof is sketched as follows. Following Theorem 8 in Chisci et al. (2001) we can show that $x_t^{[i]}$ converges to the disturbance invariant set which is included in $\bar{\mathbb{X}}_N^{[i]}$. When all agents have entered into $\bar{\mathbb{X}}_N^{[i]}$, the nominal control law $u_t^{[i]} = K_i x_t^{[i]}$ can be used since all constraints in (7) are satisfied. Then each agent Σ_i can use the knowledge of this fixed control law to predict exactly inputs of its neighbour Σ_j and thus $w_t^{[i]} = 0, \forall i \in \mathcal{I}_{1:M}$. It follows from Assumption 2.2 that states of the global system (1) converge to the origin asymptotically.

5. NUMERICAL EXAMPLE

Consider the running example in Section 2.1 with six vehicles in a platoon. The operating point of each vehicle is chosen to be $d_0 = 15$m and $v_0 = 19.44$m/s. We set up the MPC problem with sampling time $\delta_t = 50$ms and a horizon of $N = 17$. The state and input constraints are $\mathbb{X}^{[i]} = [-10.0, 10.0]\,\text{m} \times [-5.0, 5.0]\,\text{m/s}$ and $\mathbb{U}^{[i]} = [-5.0, 3.0]\,\text{m}^2/\text{s}$. Sets in (16) are selected as $\mathbb{V}^{[i]} = [-1.0, 1.0]$ and $\mathbb{T}^{[i]} = [-0.3, 0.3]$.

Figure 3 shows for each vehicle Σ_i the volume (along the positive axis) of the input tightening sets $K_i \mathbb{W}_{N-1}^{[i]}$ depending on the latency d. As can be expected from (19), the amount of tightening grows monotonically with latency. When the tightening exceeds the maximum allowable input (red dashed line), $\bar{\mathbb{U}}_k^{[i]}$ becomes empty and the problem will be infeasible. Interestingly, we observe that for a fixed latency the amount of tightening appears to converge regardless of the number of agents in the chain. This shows that our tightening approach is non-conservative. Note that Σ_2 is less tightened since it does not undergo coupling effects from its neighbour's neighbour.

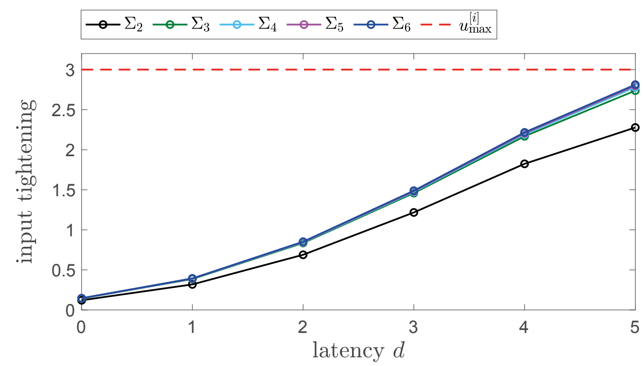

Fig. 3. Plot of the tightening set $K_i \mathbb{W}_{N-1}^{[i]}$ versus latency.

Figure 4 shows the region of attraction (RoA) of problem (7) in $e_d^{[i]}$ space as the parameter N_s varies. The RoA in this case stands for the maximal initial $e_d^{[i]}$ (with $e_v^{[i]} = 0$) such that (7) is feasible for all Σ_i. Using Algorithm 2 we design for each value of N_s a DMPC controller. Due to the difference in constraints (7d) and (7e), those controllers in general have different RoA's. Through fine tuning of the parameter N_s we observe that $N_s = 9$ yields the controller with the largest RoA (8.25m) for the example considered, since it well achieves the balance between latency and uncertainty; while in the extreme cases: $N_s = 0$ (sending nothing) and $N_s = 15$ (sending everything but with a large delay), the RoA is smaller (7.19m and 7.27m, respectively).

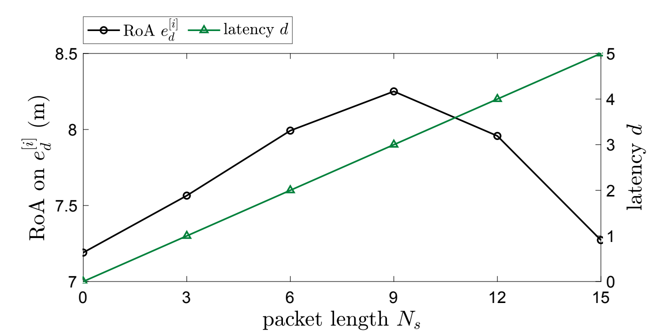

Fig. 4. Plot of N_s versus the region of attraction in terms of the relative distance $e_d^{[i]}$.

Finally Table 1 summarizes the convergence time T_c with different values of the parameter N_s, while all agents start from the same initial condition $x_0^{[i]} = [5.0, 0.0]^\top$. Again we observe that $N_s = 9$ leads to the fastest control time.

Table 1. Convergence time

Length N_s	0	3	6	9	12	15
T_c (s)	8.1	7.2	6.3	5.1	6.2	6.5

6. CONCLUSIONS

In this paper, a distributed MPC scheme is proposed which allows communication latency to be a tuning parameter. Feasibility and stability are guaranteed by leveraging robust MPC techniques with a proper constraint tightening procedure. An optimized control performance is achieved via fine tuning of the latency parameters. Future work includes generalizing this approach to a broader class of systems and treating latency d as a random variable.

REFERENCES

Chen, M., Hu, Q., Mackin, C., Fisac, J.F., and Tomlin, C.J. (2015). Safe platooning of unmanned aerial vehicles via reachability. In *2015 54th IEEE Conference on Decision and Control (CDC)*, 4695–4701. IEEE.

Chisci, L., Rossiter, J.A., and Zappa, G. (2001). Systems with persistent disturbances: predictive control with restricted constraints. *Automatica*, 37(7), 1019–1028.

Conte, C., Voellmy, N.R., Zeilinger, M.N., Morari, M., and Jones, C.N. (2012). Distributed synthesis and control of constrained linear systems. In *American Control Conference (ACC), 2012*, 6017–6022. IEEE.

Dold, J. and Stursberg, O. (2009). Distributed predictive control of communicating and platooning vehicles. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, 561–566. IEEE.

Dolk, V.S., Ploeg, J., and Heemels, W.P.M.H. (2017). Event-triggered control for string-stable vehicle platooning. *IEEE Transactions on Intelligent Transportation Systems*, 18(12), 3486–3500.

Eisen, M., Rashid, M.M., Gatsis, K., Cavalcanti, D., Himayat, N., and Ribeiro, A. (2019). Control aware radio resource allocation in low latency wireless control systems. *IEEE Internet of Things Journal*, 1–1. doi: 10.1109/JIOT.2019.2909198.

Farina, M., Perizzato, A., and Scattolini, R. (2015). Application of distributed predictive control to motion and coordination problems for unicycle autonomous robots. *Robotics and Autonomous Systems*, 72, 248–260.

Firoozi, R., Nazari, S., Guanetti, J., O'Gorman, R., and Borrelli, F. (2018). Safe adaptive cruise control with road grade preview and V2V communication. *arXiv preprint arXiv:1810.09000*.

Gatsis, K., Hassani, H., and Pappas, G.J. (2018). Latency-reliability tradeoffs for state estimation. *arXiv preprint arXiv:1810.11831*.

Hahn, J., Schoeffauer, R., Wunder, G., and Stursberg, O. (2018). Distributed MPC with prediction of time-varying communication delay. *IFAC-PapersOnLine*, 51(23), 224–229.

Hu, H., Pu, Y., Chen, M., and Tomlin, C.J. (2018). Plug and play distributed model predictive control for heavy duty vehicle platooning and interaction with passenger vehicles. In *2018 IEEE Conference on Decision and Control (CDC)*, 2803–2809. IEEE.

Jiang, X., Shokri-Ghadikolaei, H., Fodor, G., Modiano, E., Pang, Z., Zorzi, M., and Fischione, C. (2018). Low-latency networking: Where latency lurks and how to tame it. *Proceedings of the IEEE*, (99), 1–27.

Li, Y.J. (2010). An overview of the DSRC/WAVE technology. In *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, 544–558. Springer.

Maity, D., Mamduhi, M.H., Hirche, S., Johansson, K.H., and Baras, J.S. (2019). Optimal LQG control under delay-dependent costly information. *IEEE Control Systems Letters*, 3(1), 102–107.

Quevedo, D.E. and Nesic, D. (2011). Input-to-state stability of packetized predictive control over unreliable networks affected by packet-dropouts. *IEEE Transactions on Automatic Control*, 56(2), 370–375.

Scattolini, R. (2009). Architectures for distributed and hierarchical model predictive control–a review. *Journal of process control*, 19(5), 723–731.

Vukadinovic, V., Bakowski, K., Marsch, P., Garcia, I.D., Xu, H., Sybis, M., Sroka, P., Wesolowski, K., Lister, D., and Thibault, I. (2018). 3GPP C-V2X and IEEE 802.11 p for Vehicle-to-Vehicle communications in highway platooning scenarios. *Ad Hoc Networks*, 74, 17–29.

Appendix A. PROOF OF LEMMA 4.1

We start by showing $x^{[i]}_{t+1} - \bar{x}^{[i]}_{t+1|t} \in \mathbb{W}^{[i]}_1$. Introduce the shorthand notation $\Delta v^{[i]}_{t+k} = \bar{v}^{[i]}_{t+k|t+k} - \bar{v}^{[i]}_{t+k-1|t+k-1}$. First of all, notice the fact that the deviation between $x^{[1]}_t$ and $\tilde{x}^{[1]}_{t|t-d-1}$ is solely due to replanning of MPC, i.e.

$$x^{[1]}_t - \tilde{x}^{[1]}_{t|t-d-1} = \sum\nolimits_{k=1}^{d} \Phi_1^{k-1} B_{11} \Delta v^{[1]}_{t+k} \tag{A.1a}$$

$$\in \bigoplus\nolimits_{k=1}^{d} \Phi_1^{k-1} B_{11} (d+1) \mathbb{V}^{[1]} \tag{A.1b}$$

as suggested by (20b). The last step (A.1b) follows (16). For agent Σ_2, from (14) we have that

$$x^{[2]}_{t+1} - \bar{x}^{[2]}_{t+1|t} = B_{21}\left(u^{[1]}_{t|t} - \tilde{u}^{[1]}_t\right) \tag{A.2a}$$

$$= B_{21}\left[\left(v^{[1]}_{t|t} - \tilde{v}^{[1]}_{t|t-d-1}\right) + K_1\left(x^{[1]}_t - \tilde{x}^{[1]}_{t|t-d-1}\right)\right] \tag{A.2b}$$

$$\in B_{21}\left[(d+1)\mathbb{V}^{[1]} \oplus K_1 \hat{\mathbb{X}}^{[1]}_0\right] = \mathbb{W}^{[2]}_1 \tag{A.2c}$$

which is in accordance with (17b). Now, following the same idea as (A.1), agent Σ_3 is able to estimate the uncertainty in states of its neighbour Σ_2

$$x^{[2]}_t - \tilde{x}^{[2]}_{t|t-d-1} \tag{A.3a}$$

$$= \sum\nolimits_{k=1}^{d} \Phi_2^{k-1}\left[B_{22}\Delta v^{[2]}_{t+k} + B_{21}(u^{[1]}_{t-k} - \tilde{u}^{[1]}_{t-k})\right] \tag{A.3b}$$

$$\in \bigoplus\nolimits_{k=1}^{d} \Phi_1^{k-1}\left[B_{22}(d+1)\mathbb{V}^{[1]} \oplus B_{21}\hat{\mathbb{U}}^{[1]}_{d-k}\right] \tag{A.3c}$$

which coincides with (20b). By induction, (A.2) holds for all agents $\Sigma_i, \forall i \geq 2$.

The next step is to show $w^{[i]}_{t+k} = x^{[i]}_{t+k} - \bar{x}^{[i]}_{t+k|t} \in \mathbb{W}^{[i]}_k, \forall k \in \mathcal{I}_{2:N}$ as stated in (21a). Recall from (15) that

$$w^{[i]}_{t+k} = \Phi_i w^{[i]}_{t+k-1} + B_{ij}\left(u^{[j]}_{t+k-1|t} - \tilde{u}^{[j]}_{t+k-1}\right) \tag{A.4a}$$

$$\in \Phi_i \mathbb{W}^{[i]}_{k-1} \oplus B_{ij}\hat{\mathbb{U}}^{[j]}_{k-1} = \mathbb{W}^{[i]}_k \tag{A.4b}$$

which matches with (17a). The bound on $u^{[j]}_{t+k|t} - \tilde{u}^{[j]}_{t+k}$ in (A.4a) can be derived in a similar way as (A.2b).

Finally, the proof of (21b) goes as follows

$$u^{[i]}_{t+k} - \bar{u}^{[i]}_{t+k|t} = K_i\left(x^{[i]}_{t+k} - \bar{x}^{[i]}_{t+k|t}\right) \in K_i \mathbb{W}^{[i]}_k \tag{A.5a}$$

The proof is now complete.