2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Communication Demand Minimization for Perturbed Networked Control Systems with Coupled Constraints

Masoud Bahraini, Mario Zanon, Alessandro Colombo, Paolo Falcone

***Abstract*— Communication scheduling is needed when control loops of several safety-critical systems are closed through a shared communication medium. To enable schedulability, control for each system is designed primarily to minimize its communication demand. In this paper, we study communication demand minimization for a class of perturbed multi-agent networked control systems with a shared communication medium and subject to input and coupled state constraints. First, a framework to design communication schedule and control is recalled such that state and input constraints are satisfied under no coupling assumption. Then, a heuristic method is proposed to decouple state constraints such that the overall communication demand of the systems is minimized. Effectiveness of the proposed results are illustrated through a numerical example.**

I. Introduction

A networked control system (NCS) is a system whose feedback loop is closed through a communication medium. NCSs have several benefits such as reduced wiring costs, increased system agility, and eased diagnosis and maintenance. However, network imperfections such as bandwidth limitation and packet losses may degrade the control performance or cause instability [1], [2]. The communication imperfections can be taken into account explicitly to ensure the desired control performance. Another important consideration in the design is the selection of a medium access control (MAC) mechanism. There are two types of MAC: 1) random access schemes (aperiodic), and 2) scheduling schemes (periodic) [3]. While random access schemes, such as event-triggered and self-triggered control [4], may be used to minimize energy consumption or utilization of the communication medium, it is easier to guarantee performance when scheduling schemes are used [5]. Scheduling schemes are typically used in safety-critical applications, such as autonomous driving, where safety is specified through state and input constraints.

Constrained NCSs are a class of NCSs in which each system is subject to state and input constraints. Model predictive control (MPC) is used in [6] to minimize a cost function using available communication resources in a so-called token bucket network. This approach is extended in [7] using tube-based MPC and the so-called *multi-step* robust control invariant (RCI) sets to guarantee robust satisfaction of the constraints. A multi-step RCI set is invariant for a maximum of H consecutive steps under open-loop feedback. This work is further extended in [8], where output feedback is designed using an event-triggered scheme. While the token bucket network limits the communication rate, it does not trivially translate to an MAC scheme for multi-agent NCSs.

Multi-agent constrained NCSs with a shared communication medium are considered in [9], where authors define and use α, that is the longest time-interval during which the invariance is presented under open-loop feedback. The set of α of all systems is used as an instance of the so-called Pinwheel Problem (PP) to find a feasible communication schedule. This scheme is extended for general communication topologies and online schedules [10], optimal control design [11], and optimal output feedback design [12]. In these studies, state and input constraints are decoupled.

In this paper, we study the NCS class examined in [11] and extend the results to cases with coupled state constraints. As the controllers are distributed, one can choose to use either a Distributed MPC (DMPC) which takes into account the communication between controllers, or a Decentralized MPC (DeMPC) which assumes no communication between controllers, for the control design [13]. We consider a decentralized scheme, which involves no communication between the controllers. The lack of communication between the controllers introduces unavoidable conservativeness [14].

The contributions of the paper are summarized as follows: (a) control and communication scheduling design for a class of multi-agent NCSs subject to coupled constraints is formulated, (b) a solution for the formulated problem is proposed using a constraint decoupling scheme, (c) sufficient and necessary conditions for the existence of a feasible schedule are provided, and (d) a heuristic method for optimal constraint decoupling is proposed. The control and scheduling design is performed using the tools and the results from [10], [11], which are applicable with some modifications, after the constraints are decoupled. The main contributions of this paper are formulation of an optimal decoupling problem to enable schedulability and proposing a heuristic scheme to find a solution to the optimal decoupling problem.

The rest of the paper is organized as follows. In Section II several definitions and results from the literature are recalled, the control and communication scheduling design problem is formulated, a solution for the formulated problem is provided through a constraint decoupling scheme, and a second problem for optimal decoupling is formulated. In Section III necessary and sufficient conditions for the existence of a

M. Baharaini and P. Falcone are with the department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden. masoudb/falcone@chalmers.se.

M. Zanon is with IMT School for Advanced Studies Lucca, Italy. mario.zanon@imtlucca.it

A. Colombo is with DEIB, Politecnico di Milano, Italy. alessandro.colombo@polimi.it

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation.

feasible schedule for the first problem under the optimal decoupling scheme are provided. Then a heuristic method is proposed to find a solution to the optimal decoupling problem. Effectiveness of the proposed results are illustrated in Section IV through a numerical example. The paper is finally concluded in Section V.

Notation: Given two sets $\mathcal{P}, \mathcal{Q} \subset \mathbb{R}^n$, the Minkowski sum $\mathcal{P} \oplus \mathcal{Q}$ and the Pontryagin difference $\mathcal{P} \ominus \mathcal{Q}$ are defined by $\{p+q: \ p \in \mathcal{P}, \ q \in \mathcal{Q}\}$ and $\{p: \ p \oplus \mathcal{Q} \subseteq \mathcal{P}\}$, respectively. Given matrix F and set $\mathcal{P}$, set $F\mathcal{P}$ is defined as $\{Fp: \ p \in \mathcal{P}\}$. Set of integers $\mathbb{I}_a^b$ is defined as $\{a, a+1, \dots, b\}$ for arbitrary integers a and b. Cardinality of set $\mathcal{S}$ is defined as $|\mathcal{S}|$. Vector inequality $h_1 \leq h_2$ is defined element-wise. Lexicographic optimization

$$\text{lex} \max_x \ (f_1(x), \dots, f_n(x))$$

yields optimizer $x^\star$, where $f_1(x^\star) = \max_x f_1(x)$ and

$$f_i(x^\star) = \max_x f_i(x) \text{ s.t. } f_j(x) = f_j(x^\star), \ \forall j \in \mathbb{I}_1^{i-1}, \ i \in \mathbb{I}_2^n.$$

II. Preliminaries

In this section, we formulate two problems. The first problem is to design a communication schedule and control for a class of multi-agent NCSs with a shared communication medium and in the presence of coupled state and input constraints. The first problem is answered in the literature in the absence of constraint coupling. We leverage the available results and suggest a set of decoupled state constraints to solve the first problem. As the set of decoupled constraints is not unique and has an impact on schedulability, we formulate the second problem to determine an optimal set of decoupled constraints to facilitate schedulability.

A. Control and Scheduling Design Problem

Consider a set of discrete-time systems described by

$$x_i(t+1) = A_i x_i(t) + B_i u_i(t) + F_i w_i(t) \tag{1a}$$
$$x_i(t) \in \mathcal{X}_i \subset \mathbb{R}^{n_i}, \ u_i(t) \in \mathcal{U}_i, \ w_i(t) \in \mathcal{W}_i \tag{1b}$$

for $i = \mathbb{I}_1^q$, where $\mathcal{X}_i$ and $\mathcal{U}_i$ are admissible sets for state and input, and $\mathcal{W}_i$ is an admissible set for the unknown disturbance. Pair (A_i, B_i) is assumed to be controllable and admissible sets $\mathcal{X}_i$,$\mathcal{U}_i$ and $\mathcal{W}_i$ are convex polytopes which include zero in their interiors. Additionally, (1) is subject to

$$\sum_{i=1}^{q} G_i x_i \leq h, \tag{2}$$

where $G_i \in \mathbb{R}^{N \times n_i}$ and $h \in \mathbb{R}^N_{>0}$.

Consider a class of multi-agent NCSs depicted in Fig. 1. The systems, described in (1) and (2), share a communication medium. At each time instant, the communication schedule $\delta(t)$ specifies which system gets access to the communication channel. In other words, the feedback loop for each system i is closed at time instant t if and only if $\delta(t) = i$.

Problem 1. Design decentralized control policies for a set of systems described in (1) and (2) and a communication schedule for the described NCS such that all input and state constraints are satisfied for all $w_i(t) \in \mathcal{W}_i, \ i \in \mathbb{I}_1^q$ and $t \geq 0$.

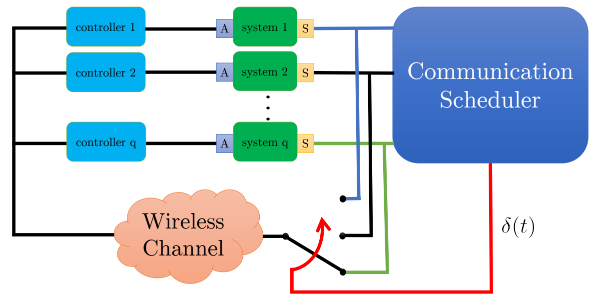

Fig. 1. Multi-agent NCS with a shared communication medium

Remark 1. One can consider Problem 1 under a more general communication topology. We refer to [10] for those topologies and skip the nuances for brevity.

B. Scheduling and Control Design for Decoupled Systems

Next, several results are recalled from the literature, to solve Problem 1 when coupled constraints (2) are omitted.

Definition 1 (Robust Control Invariant Set)**.** Set $\mathcal{C}_i$ is called a robust control invariant (RCI) set for system i, described in (1), if $\mathcal{C}_i \subseteq \mathcal{X}_i$ and

$$\forall x_i \in \mathcal{C}_i, \ \exists u_i \in \mathcal{U}_i \implies (A_i x_i + B_i u_i + F_i w_i) \in \mathcal{C}_i, \tag{3}$$

for any $w_i \in \mathcal{W}_i$.

Definition 2 (Maximal RCI Set)**.** Set $\mathcal{C}_{i,\infty}$ is called the maximal RCI (MRCI) set for system i, described in (1), if $\mathcal{C}_{i,\infty}$ is an RCI set and $\mathcal{C}_i \subseteq \mathcal{C}_{i,\infty}$, for all RCI sets $\mathcal{C}_i$.

The MRCI set for system i is its largest RCI set and a closed-loop feedback policy exists such that $x_i(t) \in \mathcal{C}_{i,\infty} \subseteq \mathcal{X}_i$ and $u_i(t) \in \mathcal{U}_i$ for all $t \geq 0$, when $x_i(0) \in \mathcal{C}_{i,\infty}$. However, condition $x_i(0) \in \mathcal{C}_{i,\infty}$ does not guarantee invariance for systems within the described NCS, since feedback loops are open at some time instants. To investigate invariance under such conditions, the open-loop behavior of the systems is studied next.

Definition 3 (Safe Time Interval)**.** Safe time interval α_i is defined as

$$\alpha_i := \max_j \{j: \ \forall x_0 \in \mathcal{C}_{i,\infty}, \ \exists u_0, \dots, u_{j-1} \in \mathcal{U}_i$$
$$\text{s.t. } A_i^j x_0 + \sum_{k=0}^{j-1} A_i^{(j-1-k)} (B_i u_k + F_i w_k) \in \mathcal{C}_{i,\infty}\} \tag{4}$$

for any disturbance $w_k \in \mathcal{W}_i, \ k \in \mathbb{I}_0^{j-1}$.

Safe time interval α_i specifies the maximum number of time instants during which state invariance can be guaranteed using an open-loop input sequence for system i, described in (1). Hence, one can guarantee state and input constraints satisfaction for system i, when the feedback loop for system i is closed at least once during any α_i consecutive time instants. Given a set of the safe time intervals $\{\alpha_i\}$, one can formulate a scheduling problem such that all feedback loops are closed frequently enough, as specified next.

Definition 4 (Pinwheel Problem (PP))**.** Find an infinite sequence $\{\delta(t)\}$ with $\delta(t) \in \mathbb{I}_1^q$ such that

$$\exists t \in \mathbb{I}_{t_0}^{t_0+\alpha_i-1} \text{ s.t. } \delta(t) = i, \ \forall i \in \mathbb{I}_1^q, \ \forall t_0 \geq 1. \tag{5}$$

The existence of a communication schedule that satisfies (5) depends on the set of safe time intervals. As α_i increases, the communication demand for system i decreases. Hence, maximization of α_i is discussed next.

Lemma 1. [from [11]] Inequality $\alpha_i(\mathcal{C}_1) \leq \alpha_i(\mathcal{C}_2)$ holds if $\mathcal{C}_2 = \gamma\mathcal{C}_1$, $\mathcal{U}_2 = \gamma\mathcal{U}_1$, and $\gamma \geq 1$, where $\mathcal{C}_1$ and $\mathcal{C}_2$ are RCI sets for system (1) and $\mathcal{U}_1$ and $\mathcal{U}_2$ are the admissible sets for the corresponding inputs.

Lemma 2. [from [11]] Assume that $\Delta\mathcal{C}$ and $\Delta\mathcal{U}$ are compact polytopes that include the origin in their interiors and

$$x_i \in \Delta\mathcal{C} \implies \exists u_i \in \Delta\mathcal{U} \text{ s.t. } A_i x_i + B_i u_i \in \Delta\mathcal{C}. \tag{6}$$

Then, inequality $\alpha(\mathcal{C}_1) \leq \alpha(\mathcal{C}_2)$ holds for $\mathcal{C}_2 = \mathcal{C}_1 \oplus \Delta\mathcal{C}$ and $\mathcal{U}_2 = \mathcal{U}_1 \oplus \Delta\mathcal{U}$, where $\mathcal{C}_1$ and $\mathcal{C}_2$ are RCI sets for system (1) and $\mathcal{U}_1$ and $\mathcal{U}_2$ are the admissible sets for the inputs.

Lemma 1 and 2 imply that α_i increases or remains the same, as the RCI set used in (4) enlarges. This is the reason why in Definition 3, the largest RCI set, i.e., $\mathcal{C}_{i,\infty}$ is used.

Remark 2. Safe time interval α_i can be defined based on any RCI set $\mathcal{C}_i$. While Lemma 1 and 2 imply that α_i is maximal in certain cases when defined based on $\mathcal{C}_{i,\infty}$, they do not prove this in general. Nevertheless, we defined α_i in (4) based on $\mathcal{C}_{i,\infty}$ set since we are not aware of any approach to find an RCI set that yields a larger α_i if such set exists.

Conjecture 1. To maximize α_i, defined in (4), one can maximize the admissible set $\mathcal{X}_i$, all other things being equal.

Remark 3. Consider safe time interval α_i as a function of $\mathcal{X}$, with an arbitrary RCI set $\mathcal{C} \subseteq \mathcal{X}$. We argue that $\alpha_i(\mathcal{X}_1) \leq \alpha_i(\mathcal{X}_2)$ holds, with $\mathcal{X}_1$ and $\mathcal{X}_2$ as admissible sets for the state x_i, if $\mathcal{X}_1 \subseteq \mathcal{X}_2$. Since any RCI set $\mathcal{C}$ for which $\alpha_i(\mathcal{X}_1)$ is maximal is also an RCI set in case of the admissible set $\mathcal{X}_2$, $\alpha_i(\mathcal{X}_2)$ is at least equal to $\alpha_i(\mathcal{X}_1)$. However, Conjecture 1 may not hold in general since the safe time interval α_i is defined based on the MRCI set $\mathcal{C}_{i,\infty}$.

The last ingredient needed to provide a solution to Problem 1 is to design a decentralized control policy for each system, given MRCI set $\mathcal{C}_{i,\infty}$, and hence α_i, for all systems and a corresponding feasible schedule. In order to design such controllers, consider the following formulation

$$\min_{\bar{x},u} \sum_{k=0}^{\alpha-1} \left(\bar{x}_k^\top Q \bar{x}_k + u_k^\top R u_k\right) + \bar{x}_\alpha^\top P_f \bar{x}_\alpha \tag{7a}$$
$$\text{s.t. } \bar{x}_0 = x_0 \in \mathcal{C}_\infty, \tag{7b}$$
$$\bar{x}_{k+1} = A x_k + B u_k, \tag{7c}$$
$$u_k \in \mathcal{U}, \tag{7d}$$
$$\bar{x}_\alpha \in \bar{\mathcal{X}}_f, \tag{7e}$$

where Q, R, and P_f are positive definite matrices with appropriate sizes and $\bar{\mathcal{X}}_f := \mathcal{C}_\infty \ominus \left(\bigoplus_{j=0}^{\alpha-1} A^j F \mathcal{W}\right)$. Note that index i is omitted for brevity.

The optimization problem (7) returns a sequence of α_i control inputs, which are successively applied to system i until the next state measurement is received. Upon receiving the state measurement, the remaining sequence of inputs is discarded, and the optimization problem is solved again using the latest state measurement, denoted by x_0. Note that this scheme guarantees recursive feasibility, see Lemma 3 from [11], and therefore, it guarantees satisfaction of state and input constraints.

C. Optimal Constraint Decoupling Formulation

The recalled scheme described in Subsection II-B does not guarantee satisfaction of the coupled state constraints (2). In order to satisfy the coupled constraints in a decentralized fashion, we decouple the constraints by

$$G_i x_i \leq h_i, \ \sum_{i=1}^{q} h_i \leq h, \ h_i \geq \mathbf{0}. \tag{8}$$

Using (8), the updated admissible set for each system i is

$$\mathcal{X}_i^{\mathrm{u}}(h_i) := \{x_i \in \mathcal{X}_i : \ G_i x_i \leq h_i\}. \tag{9}$$

Remark 4. In order to solve Problem 1, use $\mathcal{X}_i^{\mathrm{u}}$ as the state's admissible set, find a communication schedule that satisfies (5), and update control inputs by solving (7) recursively.

The selection of h_i impacts the size of the set $\mathcal{X}_i^{\mathrm{u}}(h_i)$ and hence the size of the MRCI set $\mathcal{C}_{i,\infty}$. As a result, value of the safe time interval α_i is a function of h_i. The set $\{\alpha_i\}$ in turn impacts existence of a feasible schedule, as described in (5). Therefore, to enable schedulability, one can formulate

$$\min_{\delta(1),\ldots,\delta(T_r),T_r,h_1,\ldots,h_q} \quad T_r \tag{10a}$$
$$\text{s.t.} \quad \delta(t) \in \mathbb{I}_1^q, \ T_r \in \mathbb{N}, \tag{10b}$$
$$T_r \leq \prod_{i=1}^{q} \alpha_i(h_i), \tag{10c}$$
$$\sum_{k=t}^{t+\alpha_i(h_i)-1} \eta_i(k) \geq 1, \ \forall i \in \mathbb{I}_1^q, \ \forall t \in \mathbb{I}_1^{T_r}, \tag{10d}$$
$$\eta_i(k) = \begin{cases} 1 & \text{if } i = \delta(k \bmod T_r) \\ 0 & \text{if } i \neq \delta(k \bmod T_r) \end{cases}, \tag{10e}$$
$$\sum_{i=1}^{q} h_i \leq h, \ h_i \geq \mathbf{0}, \tag{10f}$$

where $\delta(0) := \delta(T_r)$, and the optimizer $\delta^\star(1), \dots, \delta^\star(T_r^\star)$ is the periodic part of a feasible schedule, if it exists. The optimization problem (10) is formulated based on the fact that existence of a feasible schedule for the PP implies that a periodic feasible schedule also exists for the PP, see [10] for additional details on schedulability of an instance of the PP. Note that the optimization problem (10) is combinatorial and generally intractable. Hence, we propose solving an alternative optimization problem, to maximize the safe time intervals collectively, as follows:

$$\operatorname{lex} \max_{h_1,\dots,h_q} \left(\min_{i\in\mathbb{I}_1^q} \alpha_i(h_i) \right) \tag{11a}$$

$$\text{s.t.} \sum_{i=1}^{q} h_i \le h,\ h_i \ge \mathbf{0}. \tag{11b}$$

Problem 2. Solve the optimization problem (11), where $\alpha_i(h_i)$ is defined as in (4) and subject to $\mathcal{C}_{i,\infty} \subseteq \mathcal{X}_i^{\mathrm{u}}(h_i)$.

The larger each safe time interval α_i is, the less communication time-slots need to be allocated to the system i. Since the communication time-slots are limited, increase of the safe time intervals facilitates schedulability. Hence, Problem 2 is formulated to maximize the safe time intervals collectively.

Remark 5. Optimization problem (11) is primarily used as a proxy for schedulability; i.e., finding $\{h_i\}$ such that the corresponding set $\{\alpha_i\}$ is schedulable. However, there might exist cases where (11) yields a non-schedulable set $\{\alpha_i(h_i)\}$, while (10) has a feasible solution.

III. Main Results

In this section, we provide the main results that are: necessary and sufficient conditions for existence of a feasible schedule for Problem 1 under optimal constraint decoupling and a heuristic method to solve Problem 2.

Lemma 3. For any set of h_i, specified in (8), inclusion

$$\mathcal{X}_i^{\mathrm{u}}(h_i) \subseteq \mathcal{X}_i^{\mathrm{u}}(h_j), \tag{12}$$

holds for all h_j if $h_i \le h_j$.

Proof. $x \in \mathcal{X}_i^{\mathrm{u}}(h_i) \implies x \in \mathcal{X}_i^{\mathrm{u}}(h_j)$ based on (9). $\square$

Lemma 4. Inequality $\alpha_i(h_i) \le \bar{\alpha}_i$ holds, where

$$\bar{\alpha}_i := \max_t \{t :\ \mathcal{W}_i^t \subseteq \bar{\mathcal{C}}_{i,\infty}\},\ \mathcal{W}_i^t := \bigoplus_{j=0}^{t-1} A_i^j F_i \mathcal{W}_i, \tag{13}$$

and $\bar{\mathcal{C}}_{i,\infty}$ is the MRCI set for system i, described in (1), with $\mathcal{X}_i^{\mathrm{u}}(h)$ as the admissible set for the state.

Proof. Since $\mathcal{X}_i^{\mathrm{u}}(h_i) \subseteq \mathcal{X}_i^{\mathrm{u}}(h)$, inclusion $\mathcal{C}_{i,\infty} \subseteq \bar{\mathcal{C}}_{i,\infty}$ holds as well. Furthermore,

$$\mathcal{W}_i^t \subseteq \{A_i^k x_0 + \sum_{k=0}^{j-1} \left(A_i^{j-k-1} B_i u_k \right)\} \oplus \mathcal{W}_i^t, \tag{14}$$

for all $x_0 \in \mathcal{C}_{i,\infty}$ and corresponding $u_k \in \mathcal{U}_i$. Inclusion (14) and definition of $\alpha_i(h_i)$ imply that $\mathcal{W}_i^t \subseteq \mathcal{C}_{i,\infty}$, $\forall t \le \alpha_i(h_i)$. Since $\mathcal{C}_{i,\infty} \subseteq \bar{\mathcal{C}}_{i,\infty}$, one can conclude that $\mathcal{W}_i^t \subseteq \bar{\mathcal{C}}_{i,\infty}$, $\forall t \le \alpha_i(h_i)$ and thus, $\bar{\alpha}_i \ge \alpha_i(h_i)$ holds by definition. $\square$

Lemma 5. A necessary condition for existence of a feasible schedule for the described NCS is schedulability of the instance $\{\bar{\alpha}_i\}$ by PP.

Proof. Assume that there exist a set of feasible h_i and $\alpha_i(h_i)$ such that $\{\alpha_i\}$ is accepted by PP. Therefore, an infinite sequence $\{\delta(t)\}$ exists that satisfies (5) for all $\alpha_i(h_i)$. Since $\bar{\alpha}_i \ge \alpha_i$ for all i, due to Lemma 4, the same sequence of $\{\delta(t)\}$ also satisfies (5) for $\bar{\alpha}_i$. Therefore $\{\delta(t)\}$ is a feasible schedule for $\{\bar{\alpha}_i\}$, which leads to a contradiction. $\square$

One may use Lemma 5 to verify that the necessary condition is met before attempting to find any optimal decoupling through optimization. Set $\{\bar{\alpha}_i\}$ may be used to reduce computation, for instance by halting the search for an h_i that increases $\alpha_i(h_i)$, when $\alpha_i(h_i) = \bar{\alpha}_i$.

Lemma 6. A sufficient condition for existence of a feasible schedule for the described NCS is schedulability of instance $\{\alpha_i(h_i)\}$ by PP, for an arbitrary set of h_i that satisfies (11b).

Proof. Since $\{\alpha_i(h_i)\}$ is accepted by PP, then there exists an infinite sequence $\{\delta(t)\}$ that satisfies (5). Furthermore, the state admissible set for each system is decoupled as defined in (9). Therefore, one can solve optimization problem (7) to obtain the control inputs for each system. Since (7) is recursively feasible, see Lemma 3 from [11], all constraints are satisfied robustly, and $\{\delta(t)\}$ is a feasible schedule. $\square$

In order to solve the optimization problem (11), we provide a procedure based on Conjecture 1 and Lemma 3 statements. Conjecture 1 posits that in order to increase the safe time interval for a given system, one can enlarge its state admissible set, and Lemma 3 states that one can increase the admissible set for each system i by increasing the elements of the vector h_i. In order to find a suboptimal solution for the optimization problem (11), we suggest the following steps:

1) select α and adjust h_i such that $\alpha_i(h_i) = \alpha$ for all i,
2) increase α and update h_i successively as long as (11b) is respected, to maximize the minimum $\alpha_i(h_i)$,
3) if (11b) is violated, $\alpha_i(h_i)$ has to be lower than α, i.e., equal to $(\alpha - 1)$, for one or several systems,
4) select a system and fix its corresponding h_i such that $\alpha_i(h_i) = \alpha - 1$ with a minimal h_i,
5) repeated the above step until (11b) is respected,
6) repeat the max-min problem for the rest of the systems.

Remark 6. In order to increase $\alpha_i(h_i)$, one can increase h_i incrementally. Furthermore, $\alpha_i(h_i) \ge \alpha$ may be infeasible.

Remark 7. Equality $\alpha_i(h_i) = \alpha$ may hold for a range of h_i, i.e., $h_1 \le h_i < h_2$. Furthermore, since $\sum_i h_i \le h$, for a given h_i that satisfies $\alpha_i(h_i) = \alpha$, it is of interest to minimize h_i subject to $\alpha_i(h_i) = \alpha$.

Next, we specify the described heuristic method to solve Problem 2. Consider $\{h_i^0\}$ as an initial guess for solving (11), defined by

$$h_{i,j}^0 := \begin{cases} 0 & \text{if } G_{i,j} = \mathbf{0} \\ \frac{h(i)}{M_j} & \text{if } G_{i,j} \neq \mathbf{0} \end{cases} \tag{15}$$

where M_j is the number of vectors $G_{i,j} \neq \mathbf{0}$ for all $i \in \mathbb{I}_1^q$ and $G_i^\top := \begin{bmatrix} G_{i,1}^\top & \dots & G_{i,N}^\top \end{bmatrix}$, $h_i^0 := \begin{bmatrix} h_{i,1}^0 & \dots & h_{i,N}^0 \end{bmatrix}^\top$. Using $\{h_i^0\}$ as an initial guess, Algorithms 1-4 provide a heuristic method to solve the optimization problem (11).

Algorithm 1 Heuristic method for solving problem (11)

1: normalization: $G_{i,j} := \frac{G_{i,j}}{h(j)}$ and $h(j) := 1$, $\forall j \in \mathbb{I}_1^N$, $\forall i \in \mathbb{I}_1^q$
2: select a set of step sizes: e.g., $steps = \begin{bmatrix} 0.1 & 0.05 & 0.02 & 0.01 \end{bmatrix}$
3: compute h_i^0, described in (15), and $\alpha_i(h_i^0)$ for all $i \in \mathbb{I}_1^q$
4: define $\alpha = \min_i \left(\alpha_i(h_i^0)\right)$ for all $i \in \mathbb{I}_1^q$ and define $\mathcal{S} := \mathbb{I}_1^q$
5: update h_i for all $i \in \mathbb{I}_1^q$ using Algorithm 3 with h_i^0 and α as its inputs
6: **while** $|\mathcal{S}| > 1$ **do**
7: $\quad \alpha = \alpha + 1$
8: $\quad$ find h_i^{new} for all $i \in \mathcal{S}$ using Algorithm 2
9: $\quad$ **if** Algorithm 2 returns "no solution" for system k **then**
10: $\quad\quad$ set $h = h - h_k$, remove k from $\mathcal{S}$, and jump to line 8
11: $\quad$ **end if**
12: $\quad$ minimize h_i^{new} using Algorithm 3 for all $i \in \mathcal{S}$
13: $\quad$ **while** $\sum_{i \in \mathcal{S}} h_i^{\text{new}} > h$ **do**
14: $\quad\quad$ use Algorithm 4 to select one of the systems in $\mathcal{S}$, i.e., k
15: $\quad\quad$ set $h = h - h_k$, remove k from set $\mathcal{S}$
16: $\quad$ **end while**
17: $\quad h_i = h_i^{\text{new}}$ for all $i \in \mathcal{S}$
18: **end while**
19: **if** $|\mathcal{S}| = 1$ **then**
20: $\quad h_i^{\text{new}} = h$ where $i \in \mathcal{S}$, $\alpha = \alpha_i(h_i^{\text{new}})$, and
21: $\quad$ minimize h_i^{new} using Algorithm 3, and set $h_i = h_i^{\text{new}}$
22: **end if**
23: **return** $h_1, \dots, h_q$ and $\alpha_1(h_1), \dots, \alpha_q(h_q)$

Algorithm 1 is summarized as follows:

- start from an initial guess h_i and compute $\alpha_i(h_i)$,
- set α to the minimum α_i,
- find minimal h_i such that $\alpha_i(h_i) = \alpha$ for all systems,
- increase α and find h_i such that $\alpha_i(h_i) = \alpha$,
- find minimal h_i such that $\alpha_i(h_i) = \alpha$,
- if feasible h_i for achieving $\alpha_i(h_i) = \alpha$ does not exist, select one or several of the systems whose h_i is fixed and satisfies $\alpha_i(h_i) = \alpha - 1$,
- repeat increasing α for the remaining systems.

Next, Algorithms 2-4 are described.

- Algorithm 2 increases h_i, based on given step sizes, until $\alpha_i(h_i) \geq \alpha$. It returns "no solution" if such $h_i \leq h$ does not exist. One may also consider returning $h_i = h$ after line 4; while this works within the proposed procedure, it may increase the computational burden.
- Algorithm 3 lowers h_i, based on given step sizes, to find the minimal h_i such that $\alpha_i(h_i) \geq \alpha$.
- Algorithm 4 finds the system that has the highest contribution to the violated constraints. For each $h^{\text{new}}(j) > h(j)$, one inequality constraint is violated, by $\Delta_j h$ amount. Each system has a different normalized impact on this violation, specified by $\Delta_j h_i$. Note that $\Delta_j h_i > 1$ is treated the same as $\Delta_j h_i = 1$ since resetting $h_i^{\text{new}}(j)$ with $h_i(j)$ negates this constraint violation. Cost $J(i)$ is the sum of the normalized impacts, for system i, on all the violated constraints. In order to avoid violation of the constraints, the system with the highest cost is selected to be removed. This implies that the removed systems have lower α versus the remaining systems.

Algorithm 2 Find h_i^{new} such that $\alpha_i(h_i^{\text{new}}) \geq \alpha$

1: compute $\alpha_i(h)$
2: **if** $\alpha_i(h) < \alpha$ **then**
3: $\quad$ **return** "no solution"
4: **end if**
5: **for** $p = 1$ to number-of-steps **do**
6: $\quad h_i^{\text{new}} = h_i + steps(p) \times \mathbf{1}$
7: $\quad$ **if** $h_i^{\text{new}} \leq h$ **then**
8: $\quad\quad h_i = h_i^{\text{new}}$ and compute $\alpha_i(h_i^{\text{new}})$
9: $\quad\quad$ **if** $\alpha_i(h_i^{\text{new}}) < \alpha$ **then**
10: $\quad\quad\quad$ jump to line 6
11: $\quad\quad$ **end if**
12: $\quad\quad$ **return** h_i
13: $\quad$ **end if**
14: $\quad$ **for** $q = 1$ to N **do**
15: $\quad\quad$ set $h_i^{\text{new}} = h_i$ and $h_i^{\text{new}}(q) = h_i(q) + steps(p)$
16: $\quad\quad$ **if** $h_i^{\text{new}} \leq h$ **then**
17: $\quad\quad\quad h_i = h_i^{\text{new}}$ and compute $\alpha_i(h_i^{\text{new}})$
18: $\quad\quad\quad$ **if** $\alpha_i(h_i^{\text{new}}) < \alpha$ **then**
19: $\quad\quad\quad\quad$ jump to line 14
20: $\quad\quad\quad$ **end if**
21: $\quad\quad\quad$ **return** h_i
22: $\quad\quad$ **end if**
23: $\quad$ **end for**
24: **end for**
25: **return** "no solution"

Algorithm 3 minimize h_i subject to $\alpha_i(h_i) = \alpha$

1: **for** $p = 1$ to number-of-steps **do**
2: $\quad h_i^{\text{new}} = h_i - steps(p) \times \mathbf{1}$
3: $\quad$ **if** $h_i^{\text{new}} \geq \mathbf{0}$ **then**
4: $\quad\quad$ compute $\alpha_i(h_i^{\text{new}})$
5: $\quad\quad$ **if** $\alpha_i(h_i^{\text{new}}) \geq \alpha$ **then**
6: $\quad\quad\quad h_i = h_i^{\text{new}}$ and jump to line 2
7: $\quad\quad$ **end if**
8: $\quad$ **end if**
9: $\quad$ **for** $q = 1$ to N **do**
10: $\quad\quad h_i^{\text{new}} = h_i$ and $h_i^{\text{new}}(q) = h_i(q) - steps(p)$
11: $\quad\quad$ **if** $h_i^{\text{new}} \geq \mathbf{0}$ **then**
12: $\quad\quad\quad$ compute $\alpha_i(h_i^{\text{new}})$
13: $\quad\quad\quad$ **if** $\alpha_i(h_i^{\text{new}}) \geq \alpha$ **then**
14: $\quad\quad\quad\quad h_i = h_i^{\text{new}}$ and jump to line 9
15: $\quad\quad\quad$ **end if**
16: $\quad\quad$ **end if**
17: $\quad$ **end for**
18: **end for**
19: **return** h_i

Algorithm 4 find $k \in \mathcal{S}$ with highest impact on the inequality $\sum_{i \in \mathcal{S}} h_i^{\text{new}} > h$

1: $h^{\text{new}} = \sum_{i \in \mathcal{S}} h_i^{\text{new}}$
2: $J(i) = 0$ for all $i \in \mathcal{S}$
3: **for all** $j = 1$ to N **do**
4: $\quad$ **if** $h^{\text{new}}(j) > h(j)$ **then**
5: $\quad\quad \Delta_j h = h^{\text{new}}(j) - h(j)$
6: $\quad\quad \Delta_j h_i = \frac{h_i^{\text{new}}(j) - h_i(j)}{\Delta_j h}$ for all $i \in \mathcal{S}$
7: $\quad\quad J(i) = J(i) + \min(\Delta_j h_i, 1)$ for all $i \in \mathcal{S}$
8: $\quad$ **end if**
9: **end for**
10: find k such that $J(k) = \max_i J(i)$
11: **return** k

IV. Numerical Results

Consider a network of three systems described by:

$$A_i = \begin{bmatrix} 1 & 1.1 - 0.1i \\ 0 & 1 \end{bmatrix},\ F_i = B_i = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix},\ |u_i| \leq 0.5,$$

with $|x_{i,1}| \leq 5, |x_{i,2}| \leq 1, |w_1| \leq 0.13,\ |w_2| \leq 0.11,\ |w_3| \leq 0.07$, and coupled constraints

$$\begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix} \leq \begin{bmatrix} 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 \\ 0.2 & 0 & 0 & 0 & -0.2 & 0 \end{bmatrix} x \leq \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

One can compute the upper bounds for the safe time intervals, described in (13), i.e., $\bar{\alpha}_1 = 6,\ \bar{\alpha}_2 = 7,\ \bar{\alpha}_3 = 9$. Note that the upper bounds satisfy the necessary condition, specified in Lemma 5. In this case, a simple round-robin is a feasible schedule, i.e., $\boldsymbol{\delta} := 1\ 2\ 3\ 1\ 2\ 3\ \ldots$.

Next consider the initial guess, described in (15), i.e.,

$$\begin{bmatrix} h_1^0 & h_2^0 & h_3^0 \end{bmatrix} = \begin{bmatrix} 0.5 & 0.5 & 0 \\ 0 & 0.5 & 0.5 \\ 0.5 & 0 & 0.5 \end{bmatrix}$$

The corresponding safe time intervals in this case are $\alpha_1(h_1^0) = 2,\ \alpha_2(h_2^0) = 3,\ \alpha_3(h_3^0) = 6$. Note that the above initial guess does not satisfy the sufficient condition in Lemma 6. That there is no schedule that satisfies (5) for the above safe time intervals, based on exhaustive search.

Next, we provide steps taken by Algorithm 1 in order to solve (11). First, $\alpha = 2$ is selected and

$$\begin{bmatrix} h_1 & h_2 & h_3 & h \end{bmatrix} = \begin{bmatrix} 0.39 & 0.33 & 0 & 1 \\ 0 & 0.33 & 0.21 & 1 \\ 0.15 & 0 & 0.06 & 1 \end{bmatrix}.$$

Since $h_1 + h_2 + h_3 < h$, α is increased to 3, which yields

$$\begin{bmatrix} h_1 & h_2 & h_3 & h \end{bmatrix} = \begin{bmatrix} 0.52 & 0.44 & 0 & 1 \\ 0 & 0.44 & 0.28 & 1 \\ 0.27 & 0 & 0.09 & 1 \end{bmatrix}.$$

Since $h_1 + h_2 + h_3 < h$, α is increased to 4, which yields

$$\begin{bmatrix} h_1 & h_2 & h_3 & h \end{bmatrix} = \begin{bmatrix} 0.65 & 0.55 & 0 & 1 \\ 0 & 0.55 & 0.35 & 1 \\ 0.43 & 0 & 0.15 & 1 \end{bmatrix}.$$

This time $h_1(1) + h_2(1) > h(1)$, and a violation has occurred. In this case, $J = \begin{bmatrix} 0.65 & 0.55 & 0 \end{bmatrix}$, and system one is removed, i.e.,

$$h_1^\star = \begin{bmatrix} 0.52 \\ 0 \\ 0.27 \end{bmatrix}, \quad \begin{bmatrix} h_2 & h_3 & h \end{bmatrix} = \begin{bmatrix} 0.55 & 0 & 0.48 \\ 0.55 & 0.35 & 1 \\ 0 & 0.15 & 0.73 \end{bmatrix}.$$

Since the above solution still has a violation, $h_2(1) > h(1)$, cost J is recomputed, i.e., $J = \begin{bmatrix} \text{N.A.} & 1 & 0 \end{bmatrix}$. This implies that system 2 has to be eliminated as well, which results in

$$\begin{bmatrix} h_1^\star & h_2^\star \end{bmatrix} = \begin{bmatrix} 0.52 & 0.44 \\ 0 & 0.44 \\ 0.27 & 0 \end{bmatrix}, \quad \begin{bmatrix} h_3 & h \end{bmatrix} = \begin{bmatrix} 0 & 0.04 \\ 0.35 & 0.45 \\ 0.15 & 0.73 \end{bmatrix}.$$

Since only one system is remained, $h_3^\star = h$ is selected, and thus $\alpha_1(h_1^\star) = \alpha_2(h_2^\star) = 3, \quad \alpha_3(h_3^\star) = 7$. A feasible schedule for these safe time intervals is $\boldsymbol{\delta} := 1\ 2\ 3\ 1\ 2\ 3\ \ldots$.

V. Conclusions

In this paper we formulated an optimal constraint decoupling scheme in order to minimize communication demand for a class of multi-agent perturbed networked control systems with coupled state constraints. We proposed a heuristic method to solve the formulated problem and illustrated its effectiveness through a numerical example. In future works, our plan is to extend the proposed control and communication scheduling design scheme to include systems with coupled dynamics. Additionally, we aim to expand the results to cover systems with chance constraints.

References

[1] X.-M. Zhang, Q.-L. Han, and X. Yu, "Survey on recent advances in networked control systems," *IEEE Transactions on industrial informatics*, vol. 12, no. 5, pp. 1740–1752, 2015.

[2] X.-M. Zhang, Q.-L. Han, X. Ge, D. Ding, L. Ding, D. Yue, and C. Peng, "Networked control systems: A survey of trends and techniques," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 1, pp. 1–17, 2019.

[3] M. Zanon, T. Charalambous, H. Wymeersch, and P. Falcone, "Optimal scheduling of downlink communication for a multi-agent system with a central observation post," *IEEE Control Systems Letters*, vol. 2, no. 1, pp. 37–42, 2017.

[4] W. P. Heemels, K. H. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in *2012 ieee 51st ieee conference on decision and control (cdc)*. IEEE, 2012, pp. 3270–3285.

[5] W. H. Heemels, M. Donkers, and A. R. Teel, "Periodic event-triggered control for linear systems," *IEEE Transactions on automatic control*, vol. 58, no. 4, pp. 847–861, 2012.

[6] S. Wildhagen, M. A. Müller, and F. Allgöwer, "Predictive control over a dynamical token bucket network," *IEEE Control Systems Letters*, vol. 3, no. 4, pp. 859–864, 2019.

[7] S. Wildhagen and F. Allgöwer, "Rollout scheduling and control for disturbed systems via tube mpc," in *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, pp. 3145–3150.

[8] S. Wildhagen and F. Allgower, "Uncertainties and output feedback in rollout event-triggered control," *IEEE Transactions on Control of Network Systems*, 2022.

[9] A. Colombo, M. Bahraini, and P. Falcone, "Measurement scheduling for control invariance in networked control systems," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 3361–3366.

[10] M. Bahraini, M. Zanon, P. Falcone, and A. Colombo, "Scheduling and robust invariance in networked control systems," *IEEE Transactions on Automatic Control*, vol. 67, no. 6, pp. 3075–3082, 2021.

[11] M. Bahraini, M. Zanon, A. Colombo, and P. Falcone, "Optimal control design for perturbed constrained networked control systems," *IEEE Control Systems Letters*, vol. 5, no. 2, pp. 553–558, 2020.

[12] M. Bahraini, A. Colombo, M. Zanon, and P. Falcone, "Robust control invariance for networked control systems with output feedback," in *2022 IEEE 61st Conference on Decision and Control (CDC)*. IEEE, 2022, pp. 7676–7681.

[13] A. Bemporad and D. Barcelli, "Decentralized model predictive control," *Networked control systems*, pp. 149–178, 2010.

[14] S. K. Mulagaleti, A. Bemporad, and M. Zanon, "Computation of least-conservative state-constraint sets for decentralized mpc with dynamic and constraint coupling," *IEEE Control Systems Letters*, vol. 5, no. 1, pp. 235–240, 2020.