

Optimal scheduling and control for constrained multi-agent networked control systems

Masoud Bahraini¹  | Mario Zanon² | Alessandro Colombo³ | Paolo Falcone^{1,4}

¹Electrical Engineering Department, Chalmers University of Technology, Gothenburg, Sweden

²IMT School for Advanced Studies Lucca, Lucca, Italy

³Department of Electronics, Information and Bioengineering, Politecnico di Milano, Milan, Italy

⁴Engineering Department, Università di Modena e Reggio Emilia, Modena, Italy

Correspondence

Masoud Bahraini, Electrical Engineering Department, Chalmers University of Technology, Gothenburg, Sweden.
Email: masoudb@chalmers.se

Funding information

Wallenberg AI, Autonomous Systems and Software Program

Abstract

In this paper, we study optimal control and communication schedule co-design for multi-agent networked control systems, with assuming shared parallel communication channels and uncertain constrained linear time-invariant discrete-time systems. To that end, we specify the communication demand for each system using an associated robust control invariant set and reachability analysis. We use these communication demands and invariant sets to formulate tube-based model predictive control and offline/online communication schedule co-design problems. Since the scheduling part includes an infinite dimension integer problem, we propose heuristics to find suboptimal solutions that guarantee robust constraints satisfaction and recursive feasibility. The effectiveness of our approach is illustrated through numerical simulations.

KEYWORDS

measurement scheduling, networked control systems, robust invariance

1 | INTRODUCTION

A networked control system (NCS) is a set of control loops closed over a communication network. While this entails advantages such as reduced wiring, lower cost, and more flexibility and maintainability of the system, it also causes some disadvantages, such as information losses and delays, which may degrade the performance or cause instability.¹ All these issues become even more challenging in a multi-agent setting, where the communication network is shared by multiple agents and some degree of coordination is required in order to ensure that each agent obtains sufficient access to the network. In the multi-agent setting, an important imperfection of the communication link in NCSs is therefore bandwidth limitation.

In order to allocate communication resources to the agents, a medium access protocol is necessary to determine which agents are allowed to communicate through the shared network at any given time. A medium access protocol can be deterministic, stochastic, or combination of the two. Deterministic scheduling is advantageous from the control design standpoint since it enables one to provide guarantees on the system performance. However, providing strong guarantees over a wide range of operating points may cause over-provisioning. On the other hand, random access schemes, typically developed for those applications where the communication is *sporadic*, reduce the issue of over-provisioning, that is, they potentially reduce the number of communication channels and the frequency of the communication by each system. Although in a random access scheme transmissions might be postponed due to unavailability of the communication medium, the average amount of transmissions may be significantly reduced, while some overall closed-loop system

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2021 The Authors. *Optimal Control Applications and Methods* published by John Wiley & Sons Ltd.

performance, for example, \mathcal{L}_2 stability, can be enforced.² However, the use of random access schemes in safety-critical applications makes it hard to guarantee satisfaction of state and input constraints at all times.

Previous work has focused on defining ways to relate the medium access protocol specifications to system theoretic properties. An important concept is the maximally allowable transmission interval (MATI), which establishes an upper bound between two consecutive transmissions such that stability is guaranteed while accounting for the maximum possible communication delays.³ This approach has been further extended to incorporate communication constraints, time-varying transmission intervals, and time-varying delays.⁴ Additionally, the introduction of minimal allowable transmission interval⁵ and average allowable transmission interval⁶ has been proposed to further improve the existing framework. In self-triggered transmission schemes the next communication time is decided after each transmission based on a trade-off between control performance and communication utilization, using concepts similar to MATI. This approach has been applied to constrained linear systems in order to reduce transmission while guaranteeing stability and constraint satisfaction.⁷ This scheme has been further extended to deal with constrained linear systems with additive disturbance.⁸

All previously cited schemes aim to reduce the amount of transmissions while preserving stability. However, an alternative approach, based on rollout control, aims at maximizing control performance while restricting the communication utilization according to a given traffic specification (TS). In window-based TS^{9,10} the number of transmissions are limited during each window of time, and in token bucket TS¹¹⁻¹³ a dynamic restriction is introduced to limit the transmission rate. Token bucket and window-based TS schemes have been extended to constrained linear systems with additive disturbance, in which a tube-based model predictive control (MPC) is used to guarantee robust satisfaction of state and input constraints.¹³ This method relies on a robust control invariant (RCI) set that is H time-steps invariant for the system between consecutive transmissions. Therefore, the H -RCI property is a related concept to the MATI for constrained discrete-time systems with additive disturbance.

All the aforementioned studies focus on a single system and, therefore, can deal with communication medium access by assuming that the medium is available whenever the system requests it. In order to deal with multi-agent settings in which the communication medium is shared between agents, some form of coordination (i.e., a medium access protocol) needs to be introduced. A distributed dynamic schedule which gives priority to nodes which larger errors, called “maximum-error-first with try-once-discard, ϵ has been proposed as network access protocol, using bit-wise arbitration, for linear systems.¹⁴ Conflict-free transmissions for self-triggered schemes have also been proposed.¹⁵ While the first two approaches decide on communication online, offline medium access protocols have been derived by means of optimal control and schedule co-design. One possibility is to optimize the schedule and control to minimize an H_∞ cost,¹⁶ where the schedule determines which sensor may transmit through the channel at any given time. This optimization problem is solved by formulating a linear matrix inequality (LMI) and a heuristic method for scheduling to avoid combinatorial complexity. In another contribution, the exponential convergence rate of systems is optimized in the presence of random packet losses using an LMI.¹⁷ A mixed integer quadratic program over a finite time horizon is considered in Reference 18 for NCSs, where offline and online schedules are discussed using branch and bound algorithms. However, solving this optimization problem in real time might not be viable due to its combinatorial nature. This issue is addressed in Reference 19 where, given an offline schedule, a sub-optimal online schedule called optimal pointer placement is introduced which is a shifted version of the offline schedule. While these studies directly consider the medium access for NCSs, they do not account for state and input constraints.

Scheduling and control for uncertain multi-agent constrained NCSs has been addressed using the concept of *safe time interval* to provide access to the communication medium to the agents.^{20,21} The safe time interval plays a role similar to H -RCI and MATI, but is used in a scheduling framework to coordinate the access to the medium for all agents. This is done by mapping the scheduling problem into the Pinwheel Problem or the Windows Scheduling Problem (WSP), for which a feasible schedule can be found with polynomial time algorithms when a sufficient schedulability condition holds. This strategy guarantees satisfaction of the state and input constraints for all future times. This framework is able to deal with packet losses and to adapt the schedule online to improve performance.^{21,22} Additionally, tube-based MPC can be used to maximize the safe time interval for systems with additive perturbations.²³ The main limitation of these approaches is that they do not directly optimize closed-loop performance.

In this paper, we consider optimal communication schedule and control co-design for multi-agent NCSs with uncertain constrained systems and bandwidth limitation by extending previous work,²⁰⁻²³ that is, by jointly designing the control and the communication schedule through an optimization problem. This extension eliminates unnecessary performance loss due to sequential schedule and control designs in the previous work.

Because this optimization problem is prohibitively complex, we propose to approximate it by a heuristic in order to find a suboptimal solution in real time. Additionally, this framework can compensate packet loss while guaranteeing satisfaction of the constraints if the number of packet losses are bounded during a window of time.

The main contribution of this paper is a schedule and control co-design framework to optimize a given performance index for constrained uncertain multi-agent NCSs. Because the optimization problem is often intractable, we propose a heuristic with low computational burden that accounts for optimality in an approximate way while still providing strong constraint satisfaction guarantees. The proposed heuristic restricts the scheduling search space to a finite set of feasible schedules, which makes the co-design problem tractable.

The rest of this paper is organized as follows. Preliminary results are recalled in Section 2. Offline and online scheduling optimization problems are introduced and heuristics are proposed in order to find suboptimal solutions in Section 3. Some numerical examples are given in Section 4 to illustrate the proposed method and finally the paper is concluded in Section 5.

Notation. We define the set of integer numbers between a and b as

$$\mathcal{I}_a^b := \{x \in \mathbb{Z} : a \leq x \leq b\}. \quad (1)$$

We define $\delta(t) := (\delta_1(t), \delta_2(t), \dots, \delta_{n_\delta}(t))$ to be an ordered set of n_δ integers at time t , and the sequence

$$\boldsymbol{\delta} := (\delta(0), \delta(1), \delta(2), \dots). \quad (2)$$

Based on the sequence $\boldsymbol{\delta}$, we define the set of time instants

$$T_i(\boldsymbol{\delta}) := \{t : i \in \delta(t), t \geq 0\}, \quad (3)$$

which indicates the times at which the integer i is a member of the set $\delta(t)$. We call members of the set $T_i(\boldsymbol{\delta})$ as $t_{i,0}, t_{i,1}, \dots$ which are ordered increasingly, that is, $t_{i,j} < t_{i,j+1}$ for all $j \geq 0$. The sum of two polyhedra P and Q is defined as

$$P \oplus Q = \{z : z = p + q, p \in P, q \in Q\}, \quad (4)$$

and their difference as

$$P \ominus Q = \{z : z + q \in P, \forall q \in Q\}. \quad (5)$$

Set $\mathcal{W} := (\mathcal{W}_1, \dots, \mathcal{W}_q)$ is an ordered set of polytopes $\mathcal{W}_i \subset \mathbb{R}^{n_i}$ for all $i \in \mathcal{I}_1^q$. The lexicographic optimization

$$\text{lex max}_x (f_1(x), \dots, f_n(x)), \quad (6)$$

specifies the optimizer x^* that satisfies

$$f_1(x^*) = \max_x f_1(x), \quad (7)$$

and

$$f_i(x^*) = \max_x f_i(x) \quad (8)$$

$$\text{s.t. } f_j(x) = f_j(x^*), \forall j \in \mathcal{I}_1^{i-1}, \quad (9)$$

for all $i = 2, \dots, n$.

2 | PRELIMINARIES

In this section, we consider an optimal communication schedule and control design problem for a class of multi-agent NCSs.

For each system i we define the cost functions

$$\left(J_i^{\text{avg}}(x_{i,0}, \hat{x}_{i,0}, \mathbf{U}_i, \mathcal{W}_i, \delta), J_i^{\text{b}}(x_{i,0}, \hat{x}_{i,0}, \mathbf{U}_i, \mathcal{W}_i, \delta) \right) := \text{lex max}_{\mathbf{V}_i} \mathbf{J}_i(\tilde{\mathbf{X}}_i, \mathbf{U}_i), \quad (12a)$$

$$\text{s.t. } \tilde{x}_i(t+1) = \begin{cases} \bar{x}_i(t+1) & \text{if } i \in \delta(t+1) \\ A_i \tilde{x}_i(t) + B_i u_i(t) & \text{if } i \notin \delta(t+1) \end{cases}, \quad t \in \mathcal{I}_0^\infty, \quad (12b)$$

$$\bar{x}_i(t+1) = A_i \bar{x}_i(t) + B_i u_i(t) + v_i(t), \quad t \in \mathcal{I}_0^\infty, \quad (12c)$$

$$v_i(t) \in \mathcal{W}_i, \quad t \in \mathcal{I}_0^\infty, \quad (12d)$$

$$\bar{x}_i(0) = x_{i,0}, \quad (12e)$$

$$\tilde{x}_i(0) = \hat{x}_{i,0}, \quad (12f)$$

where $x_{i,0}$ and $\hat{x}_{i,0}$ are the initial state and state prediction, respectively, and

$$\mathbf{V}_i := (v_i(0), v_i(1), \dots), \quad \mathbf{U}_i := (u_i(0), u_i(1), \dots).$$

Since the disturbance sequence \mathbf{V}_i is unknown, we introduce the variables \bar{x}_i and \tilde{x}_i , which are the system state and the predicted state for the worst-case disturbance, respectively.

Note that we optimize in a lexicographic order, that is, the optimization is performed first with respect to the first element in vector \mathbf{J}_i , then the next elements are optimized in a way which does not jeopardize optimality with respect to the preceding elements, with the vector defined as

$$\mathbf{J}_i(\tilde{\mathbf{X}}_i, \mathbf{U}_i) := \left(\hat{J}_i(\tilde{\mathbf{X}}_i, \mathbf{U}_i), \sum_{t=0}^{\infty} (\tilde{x}_i(t)^\top Q_i \tilde{x}_i(t) + u_i(t)^\top R_i u_i(t) - \hat{J}_i(\tilde{\mathbf{X}}_i, \mathbf{U}_i)) \right), \quad (13)$$

$$\hat{J}_i(\tilde{\mathbf{X}}_i, \mathbf{U}_i) := \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^T (\tilde{x}_i(t)^\top Q_i \tilde{x}_i(t) + u_i(t)^\top R_i u_i(t)), \quad (14)$$

It ought to be stressed that, since we perform lexicographic optimization, the optimization of the second cost in $\mathbf{J}_i(\tilde{\mathbf{X}}_i, \mathbf{U}_i)$ includes an additional constraint enforcing that the first cost, that is, $\hat{J}_i(\tilde{\mathbf{X}}_i, \mathbf{U}_i)$, is not worse than its previously computed optimal value.

Remark 1. Optimization problem (12) yields the worst-case cost of a given communication schedule, control policy, and initial conditions for the NCS. In case $\mathcal{W}_i := \mathcal{V}_i$, the disturbance is not vanishing and the infinite horizon quadratic cost is unbounded. This is the reason why we consider lexicographic optimization. The most important objective is to minimize the average cost \hat{J}_i , which is finite if the states and inputs remain bounded. This average cost is however independent of transients, for example, any bounded deviation from the optimum over a finite time horizon averages to 0. In order to also account for transient performance, we propose to optimize the second cost, which penalizes deviations from the optimal average cost. Note that in the nominal case, that is, $\mathcal{W}_i := \{0\}$, we have $\hat{J}_i = 0$ for any stabilizing policy, while the secondary cost selects the optimal policy among all stabilizing ones.

The communication schedule and control design should guarantee satisfaction of the input and state constraints for each system. To that end, we formulate the optimal communication schedule and control design using ideas from robust min-max²⁵ and tube-based MPC.^{26,27} We define $\mathbf{U} := (\mathbf{U}_1, \dots, \mathbf{U}_q)$ and formulate the optimal communication schedule and control design problem as

$$\text{lex min}_{\mathbf{U}, \delta} \left(\sum_{i=1}^q J_i^{\text{avg}}(x_{i,0}, \hat{x}_{i,0}, \mathbf{U}_i, \mathcal{W}_i, \delta), \sum_{i=1}^q J_i^{\text{b}}(x_{i,0}, \hat{x}_{i,0}, \mathbf{U}_i, \mathcal{W}_i, \delta) \right), \quad (15a)$$

$$\text{s.t. } u_i(t) \in \mathcal{U}_i, \quad \forall i \in \mathcal{I}_1^q, t \in \mathcal{I}_0^\infty, \quad (15b)$$

$$\hat{x}_i(t) \in \mathcal{X}_i \ominus \mathcal{F}_i(t), \quad \forall i \in \mathcal{I}_1^q, t \in \mathcal{I}_0^\infty, \quad (15c)$$

$$\hat{x}_i(t+1) = \begin{cases} x_i(t+1) & \text{if } i \in \delta(t+1) \\ A_i \hat{x}_i(t) + B_i u_i(t) & \text{if } i \notin \delta(t+1) \end{cases}, \quad t \in \mathcal{I}_0^\infty, \quad (15d)$$

$$\mathcal{F}_i(t+1) = \begin{cases} \{0\} & i \in \delta(t+1) \\ A_i \mathcal{F}_i(t) \oplus \mathcal{V}_i & i \notin \delta(t+1) \end{cases}, \quad \forall i \in \mathcal{I}_1^q, t \in \mathcal{I}_0^\infty, \quad (15e)$$

$$\hat{x}_i(0) = \begin{cases} x_i(0) & \text{if } i \in \delta(0) \\ \hat{x}_{i,0} & \text{if } i \notin \delta(0) \end{cases}, \quad \forall i \in \mathcal{I}_1^q, \quad (15f)$$

$$\mathcal{F}_i(0) = \begin{cases} \{0\} & \text{if } i \in \delta(0) \\ \mathcal{F}_{i,0} & \text{if } i \notin \delta(0) \end{cases}, \quad \forall i \in \mathcal{I}_1^q. \quad (15g)$$

Constraint (15c) is imposed to the problem in order to enforce the state constraint $x_i(t) \in \mathcal{X}_i$. Set $\mathcal{F}_i(t)$ is the reachable set for the state prediction error. Indeed, one can verify that the state prediction error $e_i(t) := x_i(t) - \hat{x}_i(t)$ satisfies

$$e_i(0) \in \mathcal{F}_i(0) \Rightarrow e_i(t) \in \mathcal{F}_i(t), \quad \forall t \geq 0. \quad (16)$$

Consistently, we define $\mathcal{F}_{i,0} := \{e_i(0)\}$. Set $\mathcal{F}_i(t)$ is used to tighten the state admissible set \mathcal{X}_i in (15c) to account for the effect of the unknown disturbance on the state evolution in time. This is clarified in the following lemma.

Lemma 1. *The state constraint $x_i(t) \in \mathcal{X}_i$ and (15c) are equivalent.*

Proof. Equality $x_i(t) = e_i(t) + \hat{x}_i(t)$ holds by definition and $\mathcal{F}_i(t)$ is the reachable set for $e_i(t)$. Therefore, the constraint $x_i(t) \in \mathcal{X}_i$ is equivalent to $\hat{x}_i(t) + e_i(t) \in \mathcal{X}_i$ for all $e_i(t) \in \mathcal{F}_i(t)$. This is equivalent to $\hat{x}_i(t) \in (\mathcal{X}_i \ominus \mathcal{F}_i(t))$ by definition. ■

Problem (15), defines a joint optimal communication schedule and control design that enforces the agents' constraints (10b). Unfortunately, this optimization problem is impossible to solve since $x_i(t)$ is unknown for any $t > 0$; furthermore, even if the states were known, the problem would be very difficult to solve since the schedule δ is an infinite sequence of vectors, each containing n_δ integer variables. In order to resolve these issues, we use the following definitions.

Consider the feedback policy $u_i(t) = -K_i x_i(t)$ for the system (10). Then, one can define positively invariant sets for the system as follows.

Definition 1 (Robust positively invariant set). Set $S_i \subseteq \mathcal{X}_i$ is called robust positively invariant for system (10) under the feedback $u_i(t) = -K_i x_i(t)$ when

$$x_i(t) \in S_i \Rightarrow x_i(t+1) \in S_i, \quad -K_i x_i(t) \in \mathcal{U}_i, \quad \forall t, \quad \forall v_i(t) \in \mathcal{V}_i. \quad (17)$$

Definition 2 (Maximal robust positively invariant set). Set $S_{i,\infty}$ is the maximal robust positively invariant set for system (10) if it is robust positively invariant and $S_i \subseteq S_{i,\infty}$ holds for all robust positively invariant sets S_i .

By definition, when the system state is inside a robust positively invariant set initially, it is kept inside that set at all future times by feedback policy $u_i(t) = -K_i x_i(t)$. Additionally, invariance of S_i ensures that the system constraints are satisfied at all future times.

In order to compute the maximal robust positively invariant set, define the admissible set \mathcal{A}_i as

$$\mathcal{A}_i := \{x : x \in \mathcal{X}_i, \quad -K_i x \in \mathcal{U}_i\}, \quad (18)$$

and the one-step controllable set^{28,29} \mathcal{P}_{K_i} as

$$\mathcal{P}_{K_i}(\Omega_k, \mathcal{V}_i) := \{x : (A_i - B_i K_i)x \in (\Omega_k \ominus \mathcal{V}_i)\}. \quad (19)$$

One can compute the maximal robust positively invariant set $S_{i,\infty}$ using Algorithm 1.

Now assume that no control policy is given for system (10) and define the one-step controllable set $\mathcal{P}_{\mathcal{U}_i}(\Omega_k, \mathcal{V}_i)$ as

$$\mathcal{P}_{\mathcal{U}_i}(\Omega_k, \mathcal{V}_i) := \{x : A_i x \in (\Omega_k \ominus \mathcal{V}_i) \oplus (-B_i \mathcal{U}_i), \quad \Omega_k \ominus \mathcal{V}_i \neq \emptyset\}. \quad (20)$$

Then, one can compute control invariant sets by relying on the following definitions.

Algorithm 1. Maximal robust positively invariant set computation (see Reference 28 or algorithm 11.4 in Reference 29)

```

 $\Omega_0 \leftarrow \mathcal{X}_i, \Omega_{-1} \leftarrow \emptyset, k = 0$ 
while  $\Omega_k \neq \Omega_{k-1}$  do
   $\Omega_{k+1} \leftarrow \Omega_k \cap \mathcal{P}_{K_i}(\Omega_k, \mathcal{V}_i)$ 
   $k \leftarrow k + 1$ 
end while
return  $\Omega_k$ 

```

Algorithm 2. Maximal robust control invariant set computation (see Reference 28 or algorithm 11.5 in Reference 29)

```

 $\Omega_0 \leftarrow \mathcal{X}_i, \Omega_{-1} \leftarrow \emptyset, k = 0$ 
while  $\Omega_k \neq \Omega_{k-1}$  do
   $\Omega_{k+1} \leftarrow \Omega_k \cap \mathcal{P}_{U_i}(\Omega_k, \mathcal{V}_i)$ 
   $k \leftarrow k + 1$ 
end while
return  $\Omega_k$ 

```

Definition 3 (RCI set). Set $C_i \subseteq \mathcal{X}_i$ is called RCI for system (10) when

$$x_i(t) \in C_i \Rightarrow \exists u_i(t) \in \mathcal{U}_i : x_i(t+1) \in C_i, \forall t, \forall v_i(t) \in \mathcal{V}_i. \quad (21)$$

Definition 4 (Maximal RCI set). Set $C_{i,\infty}$ is the maximal RCI set for system (10) when $C_{i,\infty}$ is a RCI set for the system and $C_i \subseteq C_{i,\infty}$ for all RCI sets C_i .

When the system state is inside its RCI set initially, there exists a feasible control sequence which guarantees that the state remains the invariant set for all future times and therefore, the system constraints are respected. One can compute the maximal RCI set $C_{i,\infty}$ using Algorithm 2.

Sufficient conditions for finite-time termination of Algorithms 1 and 2 can be found in References 30 and 31, respectively. For system (10), Algorithm 1 terminates in a finite time when the invariant set is nonempty, the closed-loop system is stable, and \mathcal{X}_i is compact.²⁹ Likewise, Algorithm 2 terminates in a finite time when the invariant set is nonempty and sets \mathcal{X}_i and \mathcal{U}_i are compact.³¹ Note that these invariant sets may have complex shapes. In order to reduce complexity of the maximal invariant sets, one may instead use invariant sets that are approximations of the maximal ones, see Reference 32 for instance.

3 | OPTIMAL COMMUNICATION SCHEDULE AND CONTROL CO-DESIGN

In this section, we reformulate and simplify optimization problem (15). To that end, we first use reachability analysis to derive necessary and sufficient conditions for existence of a feasible control policy. Such conditions depend on the system dynamics and constraints, and impose restrictions on how often the state measurements must be communicated to the system. Therefore, these necessary and sufficient conditions on the control design result in necessary and sufficient conditions on the communication schedule design.

We note that, in general, finding any feasible schedule is very hard since the schedule is a countably infinite set of integer variables. In order to deal with this issue, we restrict the search space of the feasible schedules. This strategy simplifies the scheduling design at the expense of optimality. We provide offline and online schedules and prove recursive feasibility of the online one.

3.1 | Necessary and sufficient conditions for feasibility of the communication schedule

In this subsection, we provide a necessary and a sufficient condition for feasibility of a communication schedule, which can be used to find an approximate solution for the optimization problem (15).

Any feasible solution of the optimization problem (15) consists of a communication schedule and a set of control policies that satisfy input and state constraints (15b) and (15c). For a given communication schedule δ , a feasible control policy for system i exists only if the set $\mathcal{F}_i(t)$ can be kept small enough for all $t \geq 0$, that is, the system receives state measurement updates frequently enough. On the other hand, when the control policy is given, each system has a certain communication demand to guarantee that its state and input constraints are respected. We next formalize these observations in order to provide mathematical conditions for the existence of a feasible communication schedule and control policy.

In case of no bandwidth limitation, each system can receive its state measurement at all times. Therefore, the current state is known exactly and $\mathcal{F}_i(t) = \{0\}$. Then, existence of a feasible control policy for the system i is equivalent to existence of a RCI set for the system which includes the initial state $x_i(0)$. In case the bandwidth is limited, that is, $\mathcal{F}_i(t) \neq \{0\}$ for some $t \geq 0$, one can use reachability analysis to verify whether a feasible control policy exists. We formalize this concept by defining the safe time interval α_i as the longest period of time during which $i \notin \delta(t)$ and a feasible control policy exists for system i :

$$\alpha_i(C_i) := \max \left\{ k : \forall x_{i,0} \in C_i, \exists u_i(0), \dots, u_i(k-1) \in \mathcal{U}_i \text{ s.t. } A^k x_{i,0} + \sum_{j=0}^{k-1} A_i^j B_i u_i(k-j-1) \in C_i \ominus \left(\bigoplus_{j=0}^{k-1} A_i^j \mathcal{V}_i \right) \right\}. \quad (22)$$

Equation (22) states that, if $x_{i,0} \in C_i$ and $i \in \delta(0)$, then there exists an input sequence which guarantees $x_i(t) \in C_i$ for all $t \leq \alpha_i(C_i)$. Consequently, if C_i is robustly control invariant and if $i \in \delta(t)$ at least once during every α_i consecutive time instants, then a feasible control policy exists such that $x_i(t) \in C_i$ for all t . In the rest of the paper, we use $\alpha_i := \alpha_i(C_{i,\infty})$ where $C_{i,\infty}$ is the maximal RCI set for the system since any RCI set C_i is a subset of $C_{i,\infty}$, that is, the maximal set includes a larger set of feasible states. We remark that inequality $\alpha_i(C_{i,\infty}) \geq \alpha_i(C_i)$ typically holds. This inequality implies that using the maximal control invariant set for the computation of α_i , that is, $\alpha_i(C_{i,\infty})$, yields the longest allowable transmission interval for the system.²³

The set of safe time intervals $\{\alpha_1, \dots, \alpha_q\}$ can be used to formalize a necessary condition which needs to be satisfied by any feasible communication schedule δ . Let us define the times when the system i receives state measurements as $t_{i,j}(\delta) < t_{i,j+1}(\delta)$, such that

$$i \in \delta(t) \Leftrightarrow t = t_{i,j}(\delta), \quad j \geq 0. \quad (23)$$

Then,

$$t_{i,j+1}(\delta) - t_{i,j}(\delta) \leq \alpha_i, \quad \forall j \geq 0, i \in \mathcal{I}_1^q, \quad (24)$$

is a necessary condition for any feasible schedule δ as pointed out by the following lemma.

Lemma 2. *Any schedule δ which is feasible for optimization problem (15) satisfies (24).*

Proof. By contradiction, assume that $i \in \mathcal{I}_1^q$ and $j \geq 0$ exist such that $t_{i,j+1}(\delta) - t_{i,j}(\delta) > \alpha_i$ and for any $x_i(t_{i,j}(\delta)) \in C_{i,\infty}$, there exists an input sequence that guarantees $x_i(t_{i,j+1}(\delta)) \in C_{i,\infty}$ regardless of the unknown disturbances. This implies $\alpha_i \geq t_{i,j+1}(\delta) - t_{i,j}(\delta)$ by definition (22) which is contradictory. ■

We note that in (24) no constraint is imposed on $t_{i,0}(\delta)$ since its value depends on the initial values of the state $x_i(0)$ and the state prediction $\hat{x}_i(0)$.

In the following lemma, we specify a sufficient condition for feasibility of a communication schedule.

Lemma 3. *Assume that all systems have access to their initial states, that is, $\hat{x}_i(0) = x_i(0)$ for all $i \in \mathcal{I}_1^q$. Then, any communication schedule δ which satisfies*

$$t_{i,0}(\delta) \leq \alpha_i, \quad t_{i,j+1}(\delta) - t_{i,j}(\delta) \leq \alpha_i, \quad \forall j \geq 0, i \in \mathcal{I}_1^q, \quad (25)$$

is a feasible schedule for the optimization problem (15).

Proof. For each system i , the initial state is known and there exists a control sequence such that $x_i(t) \in C_{i,\infty}$ for all $t \leq \alpha_i$ by definition (22). Since $t_{i,0}(\delta) \leq \alpha_i$, the measured state $x_i(t_{i,0}(\delta))$ is inside the set $C_{i,\infty}$. Given $x_i(t_{i,0}(\delta)) \in C_{i,\infty}$, a control sequence starting from $t_{i,0}(\delta)$ exists such that $x_i(t) \in C_{i,\infty}$ for all $t \leq t_{i,0}(\delta) + \alpha_i$. As a result, $x_i(t_{i,1}(\delta)) \in C_{i,\infty}$ since $t_{i,1}(\delta) \leq t_{i,0}(\delta) + \alpha_i$. Therefore, there exists a control sequence such that $x_i(t) \in C_{i,\infty}$ for all $t \geq 0$ by induction. ■

Lemma 3 specifies the sufficient condition (25) for a feasible schedule. However, as pointed out before, finding the optimal communication schedule subject to this sufficient condition is very difficult since the problem is combinatorial with a countably infinite set of integer variables. We note that not only finding the optimal schedule is very difficult, but it may also be difficult to find any feasible schedule.

While finding the optimal communication schedule δ which satisfies (25) is very difficult in general, one can use available scheduling algorithms to find feasible schedules and uses these schedules to find a suboptimal schedule. To that end, we search for a feasible schedule for the instance $I := \{\alpha_1, \dots, \alpha_q\}$ of the WSP,³³ which is equivalent to searching for a schedule δ which respects (25).²¹ Necessary and sufficient conditions for existence of a feasible schedule for an instance of the WSP are based on the density $\rho(I)$, defined as

$$\rho(I) := \sum_{i=1}^q \frac{1}{\alpha_i}. \quad (26)$$

Given an instance of the WSP, inequality $\rho(I) \leq n_\delta$ is a necessary and $\rho(I) \leq 0.5n_\delta$ is a sufficient condition for the existence of a feasible schedule for the WSP.³³ There are polynomial time algorithms which find feasible schedules for instances of the WSP, for example, when $\rho(I) \leq 0.5n_\delta$ holds.^{34,35} One can also find a feasible schedule for an instance of the WSP, if it exists, by formulating an optimization problem and using mixed-integer solvers such as CPLEX, Gurobi, etc.²¹

Remark 2. If δ is a feasible schedule for the WSP, then all shifted versions of it, that is,

$$\delta_k := (\delta(k), \delta(k+1), \delta(k+2), \dots), \quad \forall k \in \mathbb{N}, \quad (27)$$

are also feasible schedules for the WSP.

Remark 2 can be used to construct a set of schedules which respect (25). In the following section, we propose a heuristic which finds a suboptimal feasible communication schedule, based on a given nonempty set of feasible schedules

$$S_\delta \subseteq \{\delta : t_{i,0}(\delta) \leq \alpha_i, t_{i,j+1}(\delta) - t_{i,j}(\delta) \leq \alpha_i, \forall j \geq 0, i \in \mathcal{I}_1^q\}. \quad (28)$$

Note that the optimality gap in each example depends on the set of feasible schedules S_δ . If one increases cardinality of this set by adding more feasible schedules to it, the optimality gap might reduce, but the computation expense increases. While it is hard to evaluate the optimality gap exactly, a rather conservative lower bound for the cost can be obtained by assuming that the communication channel is unconstrained, as we will discuss with some examples in Section 4.

3.2 | Offline communication schedule and control design

In this subsection, we assume that a feasible communication schedule is given and we formulate an optimal control design problem based on this schedule. This formulation is used in the following subsection to jointly design the communication schedule and the control policy.

Assume that a schedule δ is given which satisfies (25); furthermore, assume that the initial state for the system i is inside the maximal RCI set for that system and this initial value is known to the controller, that is, $\hat{x}_i(0) = x_i(0) \in C_{i,\infty}$. Then, one can find the optimal control policy for the system i which guarantees the state constraint satisfaction by solving

$$\left(\bar{J}_i^{\text{avg}}(x_{i,0}, \mathcal{W}_i, \delta), \bar{J}_i^{\text{b}}(x_{i,0}, \mathcal{W}_i, \delta) \right) := \text{lex min}_{\mathbf{U}_i} \left(\text{lex max}_{\mathbf{V}_i} \mathbf{J}_i(\tilde{\mathbf{X}}_i, \mathbf{U}_i) \right), \quad (29a)$$

$$\text{s.t. } \tilde{x}_i(t+1) = \begin{cases} \bar{x}_i(t+1) & \text{if } i \in \delta(t+1) \\ A_i \tilde{x}_i(t) + B_i u_i(t) & \text{if } i \notin \delta(t+1) \end{cases}, \quad t \in \mathcal{I}_0^\infty, \quad (29b)$$

$$\bar{x}_i(t+1) = A_i \bar{x}_i(t) + B_i u_i(t) + v_i(t), \quad t \in \mathcal{I}_0^\infty, \quad (29c)$$

$$\tilde{x}_i(t, x_{i,0}, \mathbf{U}_i, \mathcal{W}_i, \delta) \in C_{i,\infty} \ominus \left(\bigoplus_{j=0}^{\gamma_i(\delta,t)-1} A_i^j \mathcal{V}_i \right), \quad \forall t \geq 0, \quad (29d)$$

$$\gamma_i(\delta, t) = \begin{cases} t, & \text{if } t < t_{i,0}(\delta) \\ \min\{t - t_{i,j}(\delta) \geq 0 : j \geq 0\}, & \text{if } t \geq t_{i,0}(\delta) \end{cases}, \quad (29e)$$

$$u_i(t) \in \mathcal{U}_i, \quad \forall t \geq 0, \quad (29f)$$

$$v_i(t) \in \mathcal{W}_i, \quad t \in \mathcal{I}_0^\infty, \quad (29g)$$

$$\bar{x}_i(0) = \tilde{x}_i(0) = x_{i,0}, \quad (29h)$$

recursively at $t = 0, t_{i,0}(\delta), t_{i,1}(\delta), \dots$, where variable $\gamma_i(\delta, t)$, defined in (29e), counts how many time steps the system i has evolved open loop. In (29d), the RCI set $C_{i,\infty}$ is tightened, similarly to (15c), to account for the unknown disturbances and guarantee satisfaction of $x_i(t) \in C_{i,\infty} \subseteq \mathcal{X}_i$. The set \mathcal{W}_i can be defined as either $\{0\}$ or \mathcal{V}_i depending on the design scenario. Since the future states are unknown, the predicted state $\tilde{x}_i(t)$ is updated with the worst-case scenario trajectory $\bar{x}_i(t)$ at the update time instants.

Define \mathbf{U}_i^* as the minimizer of the above optimization problem. Then, if $x_{i,0} = \hat{x}_{i,0} \in C_\infty$ the control sequence \mathbf{U}_i^* guarantees $x_i(t) \in C_{i,\infty}$ for all $t \leq t_{i,0}$. Furthermore, the recursive solution of (29) guarantees $x_i(t) \in C_{i,\infty}$ for all $t \geq 0$, as pointed out in the following lemma.

Lemma 4. *The optimization problem (29) is recursively feasible and its recursive solution guarantees $x_i(t) \in C_{i,\infty}$ for all $t \geq 0$.*

Proof. We first prove that the optimization problem (29) is initially feasible. Inclusion $\tilde{x}_i(0) \in C_{i,\infty}$ holds by construction and since the schedule δ satisfies (25), the inequality $t_{i,0} \leq \alpha_i$ holds by assumption. Therefore, the control sequence $u_i(0), \dots, u_i(t_{i,0} - 1) \in \mathcal{U}_i$ exists that satisfies (29d) for all $t \leq t_{i,0}$ based on the definition (22). Satisfaction of (29f) and (29d) in this period and Lemma 1 imply that $\bar{x}_i(t_{i,0}) \in C_{i,\infty}$. Since $\tilde{x}_i(t_{i,0}) = \bar{x}_i(t_{i,0}) \in C_{i,\infty}$, the control sequence $u_i(t_{i,0}), \dots, u_i(t_{i,1} - 1) \in \mathcal{U}_i$ exists that satisfies (29d) for $t \leq t_{i,1}$ and $\tilde{x}_i(t_{i,1}) = \bar{x}_i(t_{i,1}) \in C_{i,\infty}$. This reasoning implies that the problem is initially feasible by induction.

The feasible solution of the optimization problem also guarantees $x_i(t) \in C_{i,\infty}$ for all $t \leq t_{i,0}$ based on definition of α_i and the assumption $\bar{x}_i(0) := x_i(0) \in C_{i,\infty}$. Assume that the input $u_i^*(0), \dots, u_i^*(t_{i,0} - 1)$ has been applied to the system, orderly; since $i \in \delta(t_{i,0})$, the state $x_i(t_{i,0})$ is measured and available to the controller. One can define $x_{i,0} := x_i(t_{i,j})$ in (29) and solve the problem recursively at time instants $t_{i,j}$ for all $j \geq 0$ which guarantees $x_i(t) \in C_{i,\infty} \subseteq \mathcal{X}_i$ for all $t \geq 0$. ■

Using the cost function (29a), one can find an optimal feasible schedule by solving

$$\text{lex min}_\delta \left(\sum_{i=1}^q \bar{J}_i^{\text{avg}}(x_i(0), \mathcal{W}_i, \delta), \sum_{i=1}^q \bar{J}_i^{\text{b}}(x_i(0), \mathcal{W}_i, \delta) \right), \quad (30a)$$

$$\text{s.t. } \delta \in S_\delta, \quad (30b)$$

where S_δ , specified in (28), is a finite set of feasible schedules for the instance $\{\alpha_1, \dots, \alpha_q\}$ of the WSP. Note that optimization problem (30) is not hard to solve since the optimal solution can be found by enumerating among members of the set S_δ . One can use the optimal solution of (30) as the feasible schedule in the optimization problem (29) in order to calculate the optimal control policy. Note that the solution of (29) needs to get updated based on the state measurements in order to guarantee $x_i(t) \in C_{i,\infty}$ for all $t \geq 0$. This scheme is summarized in Algorithm 3.

Remark 3. Note that in Algorithm 3 the schedule δ^* is computed inside the communication scheduler, see Figure 1. At time $k = 0$, each controller i has access to $x_i(0)$ by assumption, and solves the optimization problem (29) to obtain the optimal control trajectory $u_i^*(0), u_i^*(1), \dots$. Each controller i applies $u_i^*(k)$ to system i at time k . At each time $k \geq 1$, if $i \in \delta^*(k)$, then controller i receives a new state measurement, in which case it solves the optimization problem (29) with the new initial condition $x_{i,0} = x_i(k)$ and updates $u_i^*(k), u_i^*(k+1), \dots$.

Algorithm 3. Offline communication scheduling and online control design using the optimization problem (30)

Offline:

- Calculate the safe time interval α_i using (22) for all $i \in \mathcal{I}_1^q$
- Find feasible schedules for the instance $\{\alpha_1, \dots, \alpha_q\}$ of the WSP and define the set S_δ as in (28)
- Solve the optimization problem (30) to find the offline schedule δ^*
- Solve the optimization problem (29) inside the controller i which yields $u_i^*(t)$, for all $i \in \mathcal{I}_1^q$
- Pass $u_i^*(0)$ to the actuator i , for all $i \in \mathcal{I}_1^q$

Online:

- while** $k \geq 1$ **do**
 - for** $i \in \delta^*(k)$ **do**
 - Transmit $x_{i,0} = x_i(k)$ to the corresponding controller
 - Solve (29) and update $u_i^*(t)$ accordingly
 - end for**
 - Pass $u_i^*(k)$ to the actuator i , for all $i \in \mathcal{I}_1^q$
 - Assign $k \leftarrow k + 1$
 - end while**
-

Remark 4. The optimization problem (29) has several similarities and differences with the H -RCI approach for NCSs with TS.¹³ Tube-based MPC is used to design controls for an uncertain linear time-invariant discrete-time system with state and input constraints in both (29) and Reference 13. Furthermore, both approaches use invariant sets to determine acceptable upper bounds on transmission intervals, that is, α_i in this paper and H in Reference 13. However, tube-based MPC is used in (29) only to guarantee that the system remains inside the admissible set, while Reference 13 aims at asymptotically stabilizing the system to a smaller set, for example, the minimal robust positively invariant set. This difference is reflected in slight but important differences in the tube-based MPC formulations, such that the two approaches are similar in philosophy but solve different problems.

The optimal solution of (30) is an offline schedule since it is designed initially and it does not get updated afterwards based on the available information on the state of the systems. The optimal solutions of (30) and (29) return an optimized communication schedule and control which is based on a specific disturbance sequence for the system; however, the disturbance sequence is neither controllable nor known a-priori. Therefore, at each time instant the state measurements can be used to update the communication schedule and control in order to reduce the cost in (30). Note that in the optimization problem (30), we assumed that $\hat{x}_i(0) = x_i(0)$ for all $i \in \mathcal{I}_1^q$ which cannot be assumed in the following time instants and therefore, one cannot solve the optimization problem recursively for the sake of finding an communication schedule. This issue is addressed next.

3.3 | Online communication schedule and control co-design

The offline communication schedule is only optimal for the worst-case disturbance; however, the actual disturbance is in general different, such that the offline solution is suboptimal. In this subsection, in order to eliminate this deficiency, we detail how the offline communication schedule design can be used to construct an online schedule with improved performance.

In the previous subsection, the communication schedule is fixed and in order to update the control sequence, one only needs to solve (29) for n_δ of the systems at each time instant $t > 0$. However, each system i might be allowed to wait for longer than α_i time instants to receive a new measurement update without violating the constraints, depending on the disturbances that are actually injected to the system during this time. As a result, the current state measurement for each system can be used to determine whether the state measurement update for that system can be postponed. This point is clarified by the following example.

Example 1 (Online scheduling constraints). Consider a scalar system described by (10) where $A_i = B_i = 1$, $\mathcal{X}_i = \mathcal{U}_i = \{x : |x| \leq 1\}$, and $\mathcal{V} = \{x : |x| \leq 0.1\}$. In this system, $C_{i,\infty} = \mathcal{X}_i$ and, for example, $u_i(0) = -x_i(0)$ and $u_i(t) = 0$ for all $t > 0$ is a feasible control policy that preserves the invariance for a period of time. One can show that in this case, $\alpha_i = 10$ and

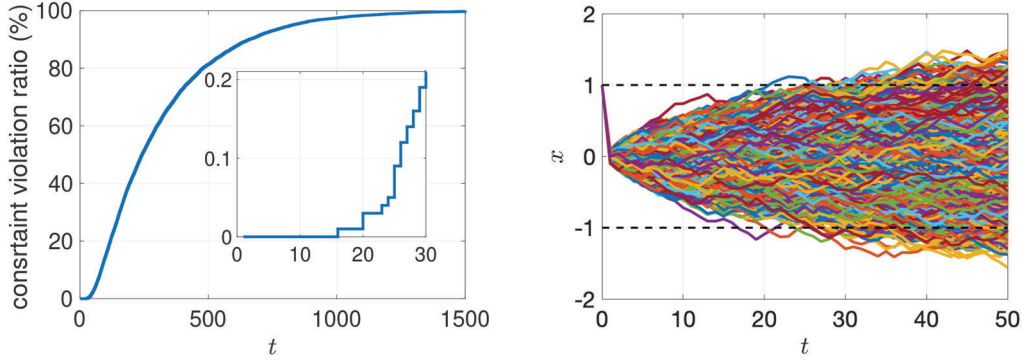


FIGURE 2 The open-loop evolution in Example 1. The system is simulated with the initial condition $x(0) = 1$ and the designed feasible control policy in the example. The disturbance sequence is assumed to be random with a uniform distribution. The system is simulated 10,000 times and the first figure shows the percentage of the number of times in which the state constraint is violated before time t . The second figure shows the state and the state constraints over time in these experiments

a feasible offline schedule should update the system i no later than $t = 10$. Now assume that at $t = 8$, the system i has not received a state measurement update yet and the state measurement shows that $x_i(8) = 0.25$. While according to the constraints at $t = 0$, the scheduler should transmit a state measurement update to the system i no later than two seconds, that is, $t \leq \alpha_i = 10$, the new measurement at $t = 8$ can be used to show that the system i remains in $C_{i,\infty}$ at least until $t \leq 15$. We have simulated this system 10,000 times with different uniformly distributed random disturbances. The simulation results in Figure 2 indicate that while $\alpha_i = 10$, in less than 0.2% of the experiments, the system needed a measurement update before $t = 30$. In this case, the average time for violating the state constraint is 317.8 which is much greater than the minimum time, that is, $\alpha_i + 1$. Nevertheless, we note that this average time might be much closer to $\alpha_i + 1$ when the system is unstable.

Example 1 illustrates that the current state measurement can be used to find less restrictive constraints on the communication schedule. Next, we formalize the state measurement update deadline for each system i based on the current state measurement. Assume that the last measurement update for the system i took place at $t_0 \leq 0$ and define $\mathbf{U}_i^* := (u_i^*(t_0), u_i^*(t_0 + 1), \dots)$ as the optimal solution to (29) solved at time t_0 for a feasible schedule δ . Furthermore, assume that the system is scheduled to receive the following state measurement update at $t_1 \geq 0$. Note that the optimal control policy cannot change before $t_1 \geq 0$ since the controller has received no additional information after time t_0 . Based on the current state measurement update $x_i(0)$, the measurement update deadline $\beta_i(x_i(0), \mathbf{U}_i^*)$, that is the time until which $x_i(t)$ is guaranteed to remain in $C_{i,\infty}$ when the previously calculated optimal control policy is applied, can be defined as

$$\beta_i := \beta_i(x_i(0), \mathbf{U}_i^*) = \max \left\{ k \in \mathbb{N} : A^k x_i(0) + \sum_{j=0}^{k-1} A^j B_i u_i^*(k-j-1) \in C_{i,\infty} \ominus \left(\bigoplus_{j=0}^{k-1} A_i^j \mathcal{V}_i \right) \right\}. \quad (31)$$

Note that $\beta_i \geq t_1$ since the control sequence \mathbf{U}_i^* guarantees $x_i(t) \in C_{i,\infty}$ for all $t_0 \leq t \leq t_1$. Therefore, one can use β_i to postpone the state measurement update for the system i and reduce the scheduling conservativeness.

We propose an online communication schedule and control co-design in which the schedule constraints are adjusted based on the current state measurements. Since the communication schedule changes, in general, at each time instant in an online communication schedule design, one needs to evaluate (29a) for all $i \in \mathcal{I}_1^q$ and at each time instant in order to pick the optimal schedule. As a result, the optimization problem (29) should be solved at each time instants, rather than only at time instants at which the system i has received a state measurement update. Therefore, next we modify (29) to be able to evaluate its cost function at all time instants. Note that the optimal control policy \mathbf{U}_i^* can not be modified during the open-loop evolution periods since the controller has received no new information. Therefore, in order to respect this limitation, we reformulate (29) as

$$\left(\bar{\mathcal{J}}_i^{\text{avg}}(\zeta), \bar{\mathcal{J}}_i^{\text{b}}(\zeta) \right) := \text{lex min}_{\mathbf{U}_i} \left(\text{lex max}_{\mathbf{V}_i} \mathbf{J}_i(\bar{\mathbf{X}}_i, \mathbf{U}_i) \right), \quad (32a)$$

$$\text{s.t. } \bar{x}_i(t+1) = \begin{cases} \bar{x}_i(t+1) & \text{if } i \in \delta(t+1) \\ A_i \bar{x}_i(t) + B_i u_i(t) & \text{if } i \notin \delta(t+1) \end{cases}, \quad t \in \mathcal{I}_0^\infty, \quad (32b)$$

$$\bar{x}_i(t+1) = A_i \bar{x}_i(t) + B_i u_i(t) + v_i(t), \quad t \in \mathcal{I}_0^\infty, \quad (32c)$$

$$\bar{x}_i(t, x_{i,0}, \mathbf{U}_i, \mathcal{W}_i, \delta) \in C_{i,\infty} \ominus \left(\bigoplus_{j=0}^{\gamma_i(\delta,t)-1} A_i^j \mathcal{V}_i \right), \quad \forall t \geq 0, \quad (32d)$$

$$\gamma_i(\delta, t) = \begin{cases} t, & \text{if } t < t_{i,0}(\delta) \\ \min\{t - t_{i,j}(\delta) \geq 0 : j \geq 0\}, & \text{if } t \geq t_{i,0}(\delta) \end{cases}, \quad (32e)$$

$$u_i(t) \in \mathcal{U}_i, \quad \forall t \geq 0, \quad (32f)$$

$$u_i(t) = u_i^{\star, \text{priv}}(t), \quad \forall t < t_{i,0}(\delta), \quad (32g)$$

$$v_i(t) \in \mathcal{W}_i, \quad t \in \mathcal{I}_0^\infty, \quad (32h)$$

$$\bar{x}_i(0) = x_{i,0}, \quad (32i)$$

$$\tilde{x}_i(0) = \hat{x}_{i,0}, \quad (32j)$$

where $\zeta := (x_{i,0}, \hat{x}_{i,0}, \mathcal{W}_i, \delta, \mathbf{U}_i^{\star, \text{priv}})$ and $\mathbf{U}_i^{\star, \text{priv}}$ is the optimal control policy calculated at the previous time instant. $\mathbf{U}_i^{\star, \text{priv}}$ is used in (32g) to preserve the latest optimal control policy until a new state measurement update arrives. In (32d) the invariant set is tightened to account for the unknown disturbances starting from the time at which the system receives a new state measurement update. Variables $x_{i,0}$ and $\hat{x}_{i,0}$ are the current state and current state prediction, respectively, which in general have different values, unlike (29).

Next, we formulate (30) using the measurement update deadline β_i defined in (31) and the cost function (32) as

$$\text{lex min}_{\delta} \left(\sum_{i=1}^q \bar{J}_i^{\text{avg}}(\zeta), \sum_{i=1}^q \bar{J}_i^{\text{b}}(\zeta) \right), \quad (33a)$$

$$\text{s.t. } \delta \in \mathcal{S}_{\delta}, \quad (33b)$$

$$t_{i,0}(\delta) \leq \beta_i(x_i(0), \mathbf{U}_i^{\star, \text{priv}}), \quad \forall i \in \mathcal{I}_1^q, \quad (33c)$$

$$\hat{x}_i(0) = \begin{cases} x_{i,0} & \text{if } i \in \delta(0) \\ \hat{x}_{i,0} & \text{if } i \notin \delta(0) \end{cases}, \quad \forall i \in \mathcal{I}_1^q, \quad (33d)$$

where $\mathbf{U}_i^{\star, \text{priv}} := u_i^{\star, \text{priv}}(-1), u_i^{\star, \text{priv}}(0), \dots$, is the optimal control policy at the previous time instant, $x_{i,0}$ is the current measured state, and $\hat{x}_{i,0}$ is defined as $A_i \hat{x}_i(-1) + B_i u_i^{\star, \text{priv}}(-1)$, that is the state prediction at the current time instant. Similarly to (30b), the constraint (33b) is considered to limit the scheduling search space and reduce the difficulty of the problem. Note that while according to the constraints (33b) and (33c) the system i is scheduled to receive a state measurement update in the $\min(\alpha_i, \beta_i)$ following time instants, this update can get postponed later on, since the following state measurements might indicate that the system does not need an update. In the optimization problem (33), the previous optimal inputs are needed, therefore, one could use (30) assuming that $\hat{x}_i(0) := x_i(0)$ at the first iteration to start the online co-design using (33) at the following time instants. This scheme is summarized in Algorithm 4.

Remark 5. Note that in Algorithm 4 the optimal schedule δ^* is updated inside the communication scheduler at each time $k \geq 1$, see Figure 1. At time $k = 0$, each controller i has access to $x_i(0)$ by assumption, and solves the optimization problem (29) to obtain the optimal control trajectory $u_i^*(0), u_i^*(1), \dots$. Each controller i applies $u_i^*(k)$ to system i at time k . At each time $k \geq 1$, problem (32) is solved inside by scheduler using the information available to the controller. This is necessary in order to define the cost to be optimized in problem (33) which yields the optimal schedule update $\delta^*(0)$. If $i \in \delta^*(0)$, then controller i also solves problem (32) with the new initial condition $\hat{x}_{i,0} = x_{i,0} = x_i(k)$ and updates $u_i^*(k), u_i^*(k+1), \dots$.

Algorithm 4. Online communication scheduling and control design using the optimization problem (33)**Offline:**

Calculate the safe time interval α_i using (22) for all $i \in \mathcal{I}_1^q$
 Find feasible schedules for the instance $\{\alpha_1, \dots, \alpha_q\}$ of the WSP and define the set S_δ as in (28)
 Solve the optimization problem (30) inside controller i which yields $u_i^*(t)$, for all $i \in \mathcal{I}_1^q$
 Assign $\hat{x}_{i,0} \leftarrow \tilde{x}_i(1)$, for all $i \in \mathcal{I}_1^q$
 Pass $u_i^*(0)$ to the actuator i , for all $i \in \mathcal{I}_1^q$

Online:

while $k \geq 1$ **do**
 Measure $x_i(k)$ and assign $x_{i,0} \leftarrow x_i(k)$ for all $i \in \mathcal{I}_1^q$
 Solve the optimization problem (33) which yields $\delta^*(t)$ and $u_i^*(t)$
 Assign $\hat{x}_{i,0} \leftarrow A_i \hat{x}_i(0) + B_i u_i^*(0)$, for all $i \in \mathcal{I}_1^q$ (to be fixed)
for all $i \in \delta^*(0)$ **do**
 Transmit $x_i(k)$ from the sensor to the controller
 Consider $\tilde{x}_i(0) = x_{i,0} = x_i(k)$, solve (32), and update $u_i^*(t)$ accordingly
end for
 Pass $u_i^*(k)$ to the actuator i , for all $i \in \mathcal{I}_1^q$
 Assign $k \leftarrow k + 1$
end while

Remark 6. Assume $\delta^b := \delta^b(0), \delta^b(1), \dots$ is a schedule which is feasible for the optimization problem (30). Then, the online communication schedule $\delta := \delta^b(k), \delta^b(k+1), \dots$ is a feasible schedule for the optimization problem (33) where k denotes the time at which (33) is solved.

Lemma 5. *Optimization problem (33) is recursively feasible and any feasible solution for this problem is also feasible for (15).*

Proof. The optimization problem is initially feasible based on Lemma 4. Consider the optimal solution of (33) as $(\hat{\mathbf{X}}^*, \mathbf{U}^*, \delta^*)$. We will prove next that this solution satisfies constraints (15b) and (15c). The constraint (15b) holds because \mathbf{U}^* respects (32f). Additionally, the constraint (33c) guarantees that each system i receives a state measurement update before $t \leq \beta_i$ which indicates $x_i(t) \in C_{i,\infty}$ for all $t \leq \beta_i$ based on (31). Once the system i received a state measurement update, which happens at $t \leq \beta_i$, Lemma 4 and (33b) imply that the solution is recursively feasible and $x_i(t) \in C_{i,\infty}$ also holds afterwards. Satisfaction of $x_i(t) \in C_{i,\infty}$ for all $t \geq 0$ in turn implies that (15c) holds based on Lemma 1. ■

Remark 7. In the optimal communication schedule and control design problems (30) and (33), one can consider a linear state feedback, that is, $u_i(t) = -K_i \tilde{x}_i(t)$, where the optimal gain K_i is the solution to the Riccati equation. In this case, one should use the maximal robust positively invariant set $S_{i,\infty}$ instead of the maximal control invariant set $C_{i,\infty}$ and calculate the corresponding safe time interval α_i and measurement update deadline β_i .

Remark 8. Assume that there exists a perfect schedule for instance $\{\alpha_1, \dots, \alpha_q\}$, that is, a schedule in which each system communicates through a single channel. Then, every system can be assigned to a communication channel. In the presence of n_δ communication channels, this implies that optimization problem (33) can be separated into n_δ smaller disjoint optimization problems, that is, one for each channel.

Remark 9. The proposed joint communication schedule and control design in the optimization problems (30) and (33) can also cover packet loss under the assumption that at most n_i^L packets could be lost for system i over any time interval of length α_i . Consider, for example, the case of two systems with $\alpha_1 = 3$ and $\alpha_2 = 5$, and assume that at most one packet may be lost during each four consecutive transmissions. Then we have $n_1^L = 1$, since $\alpha_1 < 4$; and $n_2^L = 2$, since $\alpha_2 < 2 \cdot 4$.

Under this assumption for the packet loss distribution, schedulability of (30) and (33) is equivalent to existence of a schedule for instance $I = \{\tilde{\alpha}_1, \dots, \tilde{\alpha}_q\}$ of the WSP where $\tilde{\alpha}_i := \alpha_i - n_i^L$, for all i .²¹ In (30), the strategy consists in updating the schedule such that the systems which lost a packet are immediately rescheduled for communication, that is, the schedule is shifted back by one time instant whenever a packet is lost. Since in the absence of packet losses each controller i is scheduled to receive a state measurement at least once during any $\tilde{\alpha}_i$ consecutive time instants, and this transmission may

be postponed at most by n_i^L time instants in the shifted schedule, one has the guarantee that a measurement is received at least once every α_i consecutive time instants. In turn, this guarantees that the state remains inside the admissible set.

While the modified version (30) guarantees feasibility also in case of packet losses, it does not account for optimality. Similarly to the case of no packet losses, one can then use a modified version of (33), where δ is defined using $\tilde{\alpha}_i$ instead of α_i , and the schedule is updated by solving (33) using $\tilde{\beta}_i := \beta_i - m_i^L(k)$ instead of β_i , where $m_i^L(k)$ is the number of packets that can be lost over the time interval $[k, k + \beta_i - 1]$, given the packets lost in the past. In order to clarify this last aspect, consider the case of two systems having $\beta_1 = 3$ and $\beta_2 = 5$ at a given time instant k ; assume that at most one packet may be lost during each four consecutive transmissions, and assume that only one packet was lost at time $k - 1$. Then we have $m_1^L = 0$, since having lost a packet at time $k - 1$ implies that no packet can be lost in the interval $[k, k + 2] = [k, k + \beta_1 - 1]$. Similarly, one can verify that $m_2^L = 1$, since at most one packet can be lost in the interval $[k + 3, k + 7]$, which entails that at most one packet can be lost in the interval $[k, k + 4] = [k, k + \beta_2 - 1]$.

4 | NUMERICAL SIMULATION

In this section, we provide several numerical examples to show and compare the effectiveness of the proposed optimal scheduling and control design techniques.

In Example 2, first we compare the communication schedule and control obtained from the optimization problems (30) and (33) when the control is computed by a static feedback gain. Then, we compare these designs when the control is optimized as per (32). In Example 3, we increase the disturbance levels and repeat the comparison between (30) and (33) for the optimized control case. In Example 4, we compare the solutions of the optimization problems (30) and (33) when the control is derived to minimize the cost for the worst-case disturbance.

Example 2 (Networked control vehicles). Consider five remotely controlled vehicles, each considered as a point mass with a double integrator model as

$$x_i(t + 1) = A_i x_i(t) + B_i u_i(t) + F_i v_i(t), \quad (34)$$

with

$$A_i = \begin{bmatrix} 1 & 0.1 \\ 0 & 1 \end{bmatrix}, \quad B = F = \begin{bmatrix} 0.005 \\ \kappa_i \end{bmatrix}, \quad \kappa_i = 0.1i, \quad \forall i \in \mathcal{I}_1^5, \quad (35)$$

where $x_i(t)$ contains the position and the speed of the vehicle, $u_i(t)$ is the input acceleration, and $v_i(t)$ is a matched disturbance. Consider $\mathcal{U}_i = \{u \in \mathbb{R} : |u| \leq 4\}$ and $\mathcal{V}_i = \{F_i v \in \mathbb{R}^2 : |v| \leq v_i\}$ with $(v_1, \dots, v_5) = (0.90, 0.70, 0.50, 0.28, 0.22)$ as admissible sets for the inputs and the disturbances, respectively. Furthermore, consider the admissible sets for the states as

$$\mathcal{X}_i = \left\{ x \in \mathbb{R}^3 : \begin{bmatrix} -1 \\ -5 \end{bmatrix} \leq x \leq \begin{bmatrix} 1 \\ 5 \end{bmatrix} \right\}. \quad (36)$$

and the cost function matrices

$$Q_i = \begin{bmatrix} 5 & 0 \\ 0 & 1 \end{bmatrix}, \quad R_i = 0.1, \quad \forall i \in \mathcal{I}_1^5. \quad (37)$$

Also assume that only one vehicle can communicate at each time instant, that is, $n_\delta = 1$. We consider zero as the initial state and simulate the NCS for $t = 0, \dots, 500$.

First, consider $u_i(t) = -K_i \tilde{x}_i(t)$ as an additional constraint to the optimization problems (29) and (32) where K_i is the optimal feedback gain from the corresponding Riccati equation, see Remark 7. In this case, the safe time intervals are

$$\{\alpha_1, \dots, \alpha_5\} = \{4, 5, 6, 9, 10\}, \quad (38)$$

and a feasible cyclic schedule for the WSP with instance $\{4, 5, 6, 9, 10\}$, is

$$\delta^1 := 1, 2, 3, 4, 1, 2, 3, 5, \dots \quad (39)$$

Since δ^1 has period 8, one can create a set of feasible schedules as $S_\delta^1 := \{\delta_1^1, \delta_2^1, \dots, \delta_8^1\}$, where we define $\delta_k^1 := \delta^1(k), \delta^1(k+1), \dots$ as a shifted version of δ^1 for any $k \in \mathcal{I}_1^8$, such that $\delta_1^1 = \delta^1$.

In order to investigate sensitivity of the proposed approach to the set of the set of feasible schedules, we also consider

$$\delta^2 := 1, 2, 3, 1, 4, 5, 2, 1, 3, 4, 1, 2, 5, 3, \dots, \quad (40)$$

as another cyclic feasible schedule with period 14 for the problem and the set of the feasible schedules $S_\delta^2 := \{\delta_1^2, \delta_2^2, \dots, \delta_{14}^2\}$, that is the set of shifted versions of δ^2 .

Next, we provide simulation results for five cases:

- **Case 1:** solving (30) when $S_\delta := S_\delta^1$;
- **Case 2:** solving (33) when $S_\delta := S_\delta^1$;
- **Case 3:** solving (30) when $S_\delta := S_\delta^2$;
- **Case 4:** solving (33) when $S_\delta := S_\delta^2$;
- **Case 5:** an ideal communication scenario where all vehicles are always connected;

The simulation results are shown in Figure 3. In these simulations, the disturbances are generated as uniformly distributed random values and kept the same in different cases for a fair comparison.

Since the cost functions in (29) and (32) are defined based on the predicted state, in the figure we represent a quadratic cost based on the actual states, that is,

$$\text{quadratic cost} = \sum_{k=0}^t \sum_{i=1}^5 x_i(k)^\top Q_i x_i(k) + u_i(k)^\top R_i u_i(k). \quad (41)$$

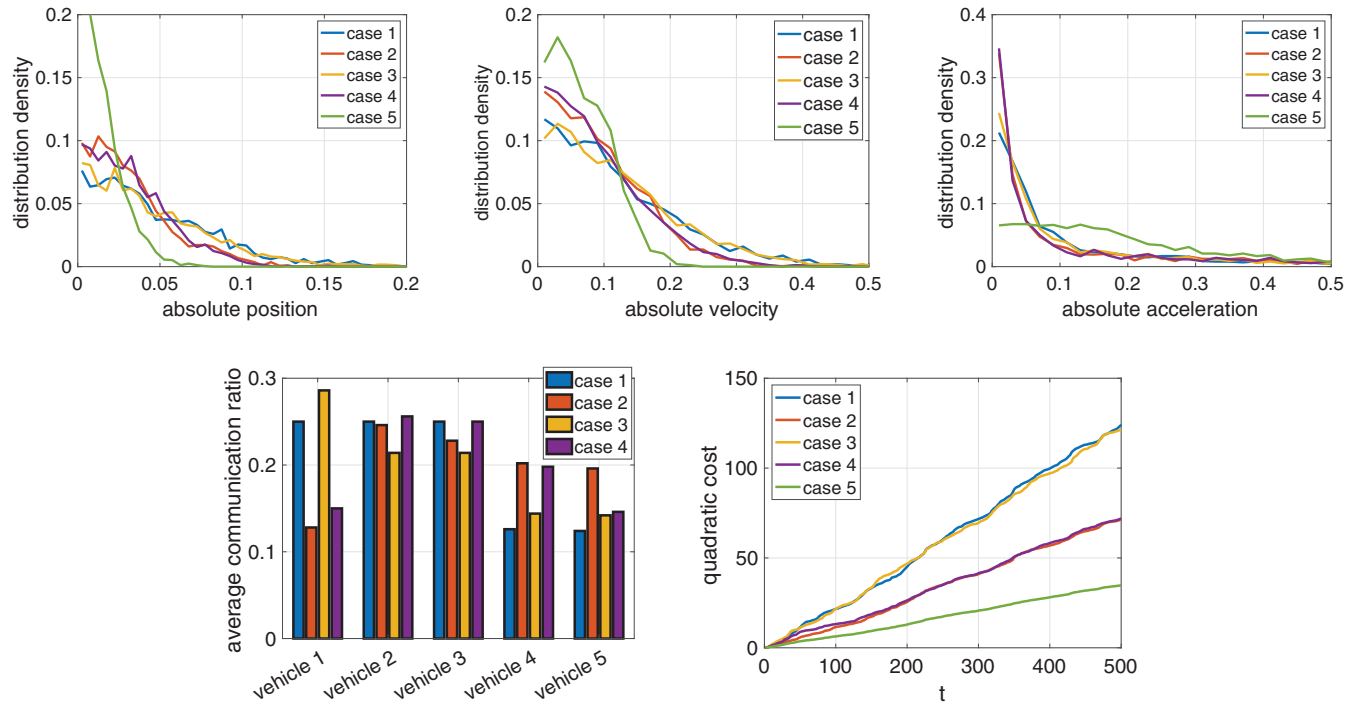


FIGURE 3 Simulation of the vehicles for different cases in Example 2 where the control is designed based on a nonvarying feedback gain; the states and inputs of all vehicles are augmented and the distribution of their absolute values are shown in the first row; the second row shows the average communication ratio for the simulated cases and the cost function for each case

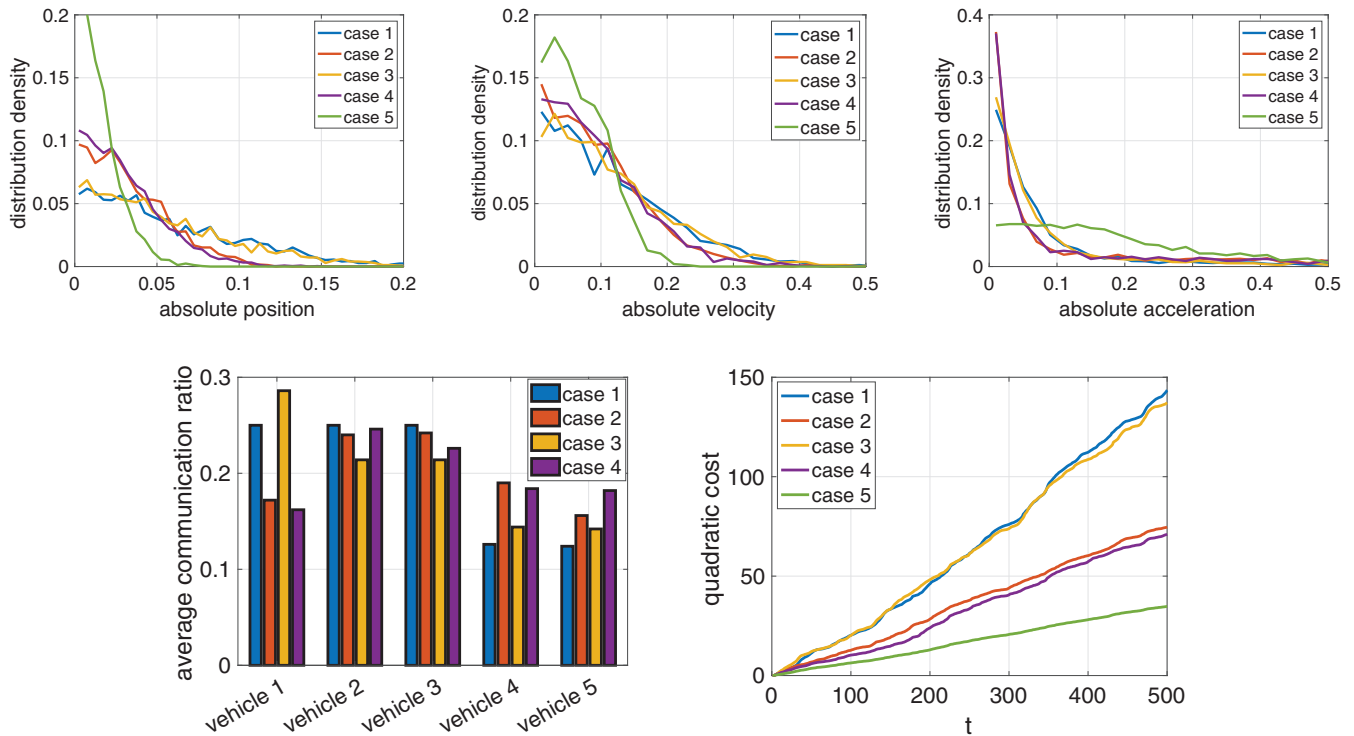


FIGURE 4 Simulation of the vehicles for different cases in Example 2 where the control is designed based on the MPC; the states and inputs of all vehicles are augmented and the distribution of their absolute values are shown in the first row; the second row shows the average communication ratio for the simulated cases and the cost function for each case

This figure indicates that the cost function for the offline schedules, that is, Cases 1 and 3, are much higher than the cost for the corresponding online schedules, that is, Cases 2 and 4. As expected, the cost for Case 5 is better than the other cases since the scheduling limitation is not present in this scenario. However, the performance of the online schedule is not dramatically worse, considering that communication is reduced by a factor 5. The distribution of the absolute value of the states of the vehicles and the inputs show that the constraints are respected; furthermore, it indicates that the performance of the offline and online schedules are not very sensitive to the schedule or the set of schedules that were used in the scheduling design.

Now consider the same NCS without considering $u_i(t) = -K_i \tilde{x}_i(t)$ as an additional constraint in the optimization problems (29) and (32), which we call the MPC design. In this case, the safe time intervals are

$$\{\alpha_1, \dots, \alpha_5\} = \{11, 9, 9, 11, 12\}. \quad (42)$$

Since the safe time intervals have increased in this case, δ^1 and δ^2 are feasible offline schedules for the network. The simulation results for the same cases are provided in Figure 4 where we have considered $\mathcal{W}_i = \{0\}$ for all $i \in \mathcal{I}_1^5$. In this simulation, the disturbances are the same as in the previous one for sake of comparison.

Similar to the Figure 3, the cost function in Figure 4 shows that the cost function for the online schedules, that is, Cases 2 and 4, are improved in comparison to the offline schedules, that is, Cases 1 and 3. Note that the cost function in the MPC cases are rather similar to ones in the constant-gain state feedback cases.

Example 3. In order to have a more critical scenario for the MPC cases, consider the previous example with $(v_1, \dots, v_5) = (2.5, 1.6, 0.9, 0.4, 0.3)$ as the disturbance bounds which results in the safe time intervals

$$\{\alpha_1, \dots, \alpha_5\} = \{4, 5, 6, 9, 10\}. \quad (43)$$

We consider zero as the initial state and simulate the NCS for $t = 0, \dots, 500$. The disturbances are generated randomly with uniform distributions and reused in different cases for sake of comparison. Since these safe time intervals are the

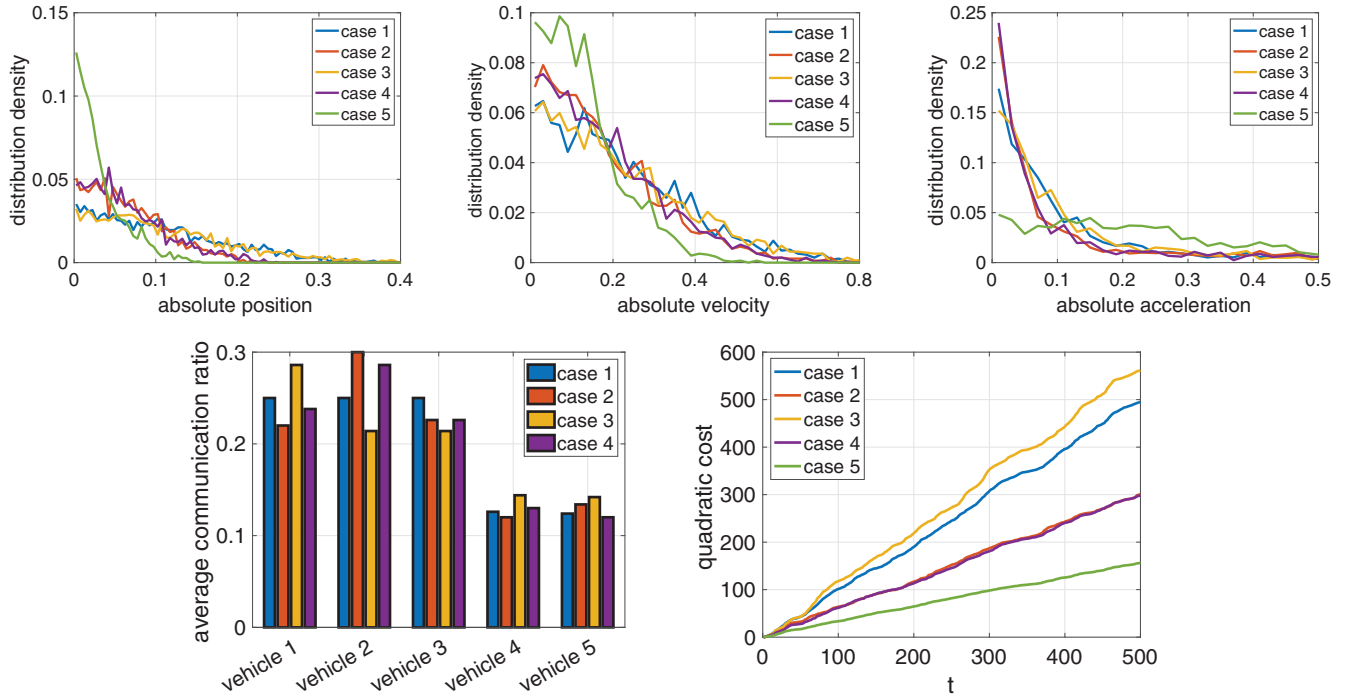


FIGURE 5 Simulation of the vehicles for different cases in Example 3 where the control is designed based on the model predictive control; the states and inputs of all vehicles are augmented and the distribution of their absolute values are shown in the first row; the second row shows the average communication ratio for the simulated cases and the cost function for each case

same as the cases with the constant feedback gain in the previous example, the same schedules are also feasible in this example. Note that in this case, the linear feedback results in lower or zero safe time intervals which results in an infeasible scheduling problem.

The simulation results are presented in Figure 5.

As in the previous example, this figure shows that the online schedules, that is, Cases 2 and 4, have better performances than the offline schedules, that is, Cases 1 and 3. The increase of the noise levels has increased the costs. One can also observe that the average communication ratio in Cases 2 and 4 for vehicle 1 have increased in comparison to the previous example since the noise level for this vehicle is increased more than other ones.

In the following, we provide an example to illustrate the results in case $\mathcal{W}_i \neq \{0\}$. Since this optimization problem is inherently difficult to solve, we have selected an extremely simple case which allows us to easily interpret the results. We also note that, since min-max problems are very difficult to solve, for nontrivial examples the infinite horizon optimization problem is often replaced by a finite horizon one. In addition, even for finite horizons, it is customary to rely on formulations which only approximately solve the min-max problem. While a thorough discussion on these methods is out of the scope of this paper, we only remind that any such formulation can be used in combination with our approach.

Example 4. Consider a network with three systems described by

$$x_i(t+1) = x_i(t) + u_i(t) + v_i(t), \quad (44)$$

where $x_i(t) \in \mathcal{X}_i = \{x : |x| \leq 1\}$, $u_i(t) \in \mathcal{U}_i = \{u : |u| \leq 1\}$, and $v_i(t) \in \mathcal{V}_i = \{v : |v| \leq v_i\}$ with $v_1 = 0.4$, $v_2 = 0.22$, and $v_3 = 0.18$. We consider zero as the initial state and simulate the NCS for $t = 0, \dots, 500$. The disturbances are generated randomly with uniform distributions and reused in different cases for sake of comparison. In this example, $C_{i,\infty} = X_i$ and $\alpha_1 = 2$, $\alpha_2 = 4$, and $\alpha_3 = 5$. One can verify that

$$\delta^3 := 1, 2, 1, 3, \dots, \quad (45)$$

is a feasible offline schedule for the optimization problem. Assume that $\mathcal{W}_i = \mathcal{V}_i$ and consider

$$\mathcal{S}_\delta^3 := \{\delta_0^3, \delta_1^3, \delta_2^3, \delta_3^3\}, \quad (46)$$

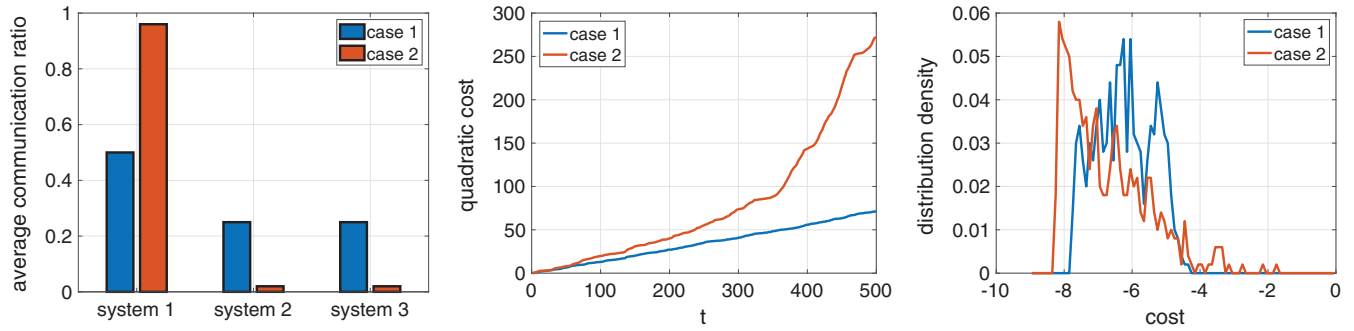


FIGURE 6 Simulation of the vehicles for different cases in Example 4; the first figure from the left shows the average communication ratio, the second figure shows the quadratic cost, as in (41), and the last figure shows the distribution of the worst-case cost, as defined in (48). The means of the worst-case cost for Cases 1 and 2 are -6.2291 and -6.7520 , respectively

as the set of feasible schedules. For simplicity, consider $Q_i = 1$ and $R_i = 0$ for all $i = 1, 2, 3$; this implies that the optimal control law, using either MPC or LQR, is

$$u_i^*(t) = \begin{cases} -x_i(t) & \text{if } \delta_i(t) = i, \\ 0 & \text{otherwise.} \end{cases} \quad (47)$$

For any $\delta \in \mathcal{S}_\delta^3$, one can verify that $\bar{J}_1^{\text{avg}} = \bar{J}_1^{\text{avg}} = 0.8$, $\bar{J}_2^{\text{avg}} = \bar{J}_2^{\text{avg}} = 1.452$, and $\bar{J}_3^{\text{avg}} = \bar{J}_3^{\text{avg}} = 0.972$. Next, we provide simulation results for two cases:

- **Case 1:** solving (30) when $\mathcal{S}_\delta := \mathcal{S}_\delta^3$;
- **Case 2:** solving (33) when $\mathcal{S}_\delta := \mathcal{S}_\delta^3$;

Figure 6 shows the simulation results for the offline and the online schedules. The worst-case cost in the figure is defined as

$$\text{worst-case cost} = \sum_{k=i}^{\infty} \sum_{i=1}^3 (\tilde{x}_i(k))^T Q_i \tilde{x}_i(k) - \bar{J}_i^b. \quad (48)$$

Note that the quadratic cost in Case 1 is less than the cost in Case 2. This is expected since in Case 2, the online scheduling aims to minimize the cost in the worst-case scenario. The worst-case cost shows that the online schedule is in fact better than the offline one. One can also observe that the online schedule communicates more often with system 1; this results in longer open-loop periods for systems 2 and 3.

5 | CONCLUSIONS

In this paper, in order to jointly design communication schedule and control for constrained NCSs, we formulated an optimization problem with a quadratic cost function. The joint design of the schedule and control prevents unnecessary performance loss of disjoint design schemes. On the one hand, this scheme exploits the periodic side of the designed communication schedule to guarantee satisfaction of the input and state constraints and on the other hand, it adjusts the design based on the state measurements which improves the performance.

One can extend this work by considering a more general network in which both the sensor-controller and the controller-actuator links are constrained and independent. Furthermore, the proposed framework could be extended to the case of systems with coupled constraints/dynamics.

ACKNOWLEDGEMENTS

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

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Masoud Bahraini  <https://orcid.org/0000-0003-0680-9655>

REFERENCES

1. Hespanha JP, Naghshtabrizi P, Xu Y. A survey of recent results in networked control systems. *Proc IEEE*. 2007;95(1):138-162.
2. Lemmon M. *Event-Triggered Feedback in Control, Estimation, and Optimization*. Springer; 2010:293-358.
3. Heemels WP, Nešić D, Teel AR, van de Wouw N. Networked and quantized control systems with communication delays. Proceedings of the 48th IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference; 2009:7929-7935.
4. Heemels WMH, Teel AR, Van de Wouw N, Nešić D. Networked control systems with communication constraints: tradeoffs between transmission intervals, delays and performance. *IEEE Trans Automat Contr*. 2010;55(8):1781-1796.
5. Heijmans SH, Postoyan R, Nešić D, Heemels WMH. Computing minimal and maximal allowable transmission intervals for networked control systems using the hybrid systems approach. *IEEE Control Syst Lett*. 2017;1(1):56-61.
6. Heijmans SH, Postoyan R, Nesic D, Heemels W. An average allowable transmission interval condition for the stability of networked control systems. *IEEE Trans Automat Contr*. 2020.
7. Berglind JB, Gommans T, Heemels W. Self-triggered MPC for constrained linear systems and quadratic costs. *IFAC Proc Vol*. 2012;45(17):342-348.
8. Brunner FD, Heemels M, Allgöwer F. Robust self-triggered MPC for constrained linear systems: a tube-based approach. *Automatica*. 2016;72:73-83.
9. Antunes D, Heemels W. Rollout event-triggered control: beyond periodic control performance. *IEEE Trans Automat Contr*. 2014;59(12):3296-3311.
10. Gommans T, Theunisse T, Antunes D, Heemels W. Resource-aware MPC for constrained linear systems: two rollout approaches. *J Process Control*. 2017;51:68-83.
11. Linsenmayer S, Allgöwer F. Performance oriented triggering mechanisms with guaranteed traffic characterization for linear discrete-time systems. Proceedings of the 2018 European Control Conference (ECC); 2018:1474-1479.
12. Wildhagen S, Müller MA, Allgöwer F. Predictive control over a dynamical token bucket network. *IEEE Control Syst Lett*. 2019;3(4):859-864.
13. Wildhagen S, Allgöwer F. Rollout scheduling and control for disturbed systems via tube MPC. Proceedings of the 2020 59th IEEE Conference on Decision and Control (CDC); 2020:3145-3150.
14. Walsh GC, Ye H. Scheduling of networked control systems. *IEEE Control Syst Mag*. 2001;21(1):57-65.
15. Henriksson E, Quevedo DE, Peters EG, Sandberg H, Johansson KH. Multiple-loop self-triggered model predictive control for network scheduling and control. *IEEE Trans Control Syst Technol*. 2015;23(6):2167-2181.
16. Lu L, Xie L, Fu M. Optimal control of networked systems with limited communication: a combined heuristic and convex optimization approach. 42nd IEEE International Conference on Decision and Control. Maui, HI: IEEE; 2003:1194-1199. https://ieeexplore.ieee.org/abstract/document/1272770?casa_token=zFopEpqshAEAAAAA:uh3YaOqjYRwE4wLtsIZP01YpL4sy0bjb4c sbaGtYBg8PePYxxYLKyJB6Jf4La9uP4GpIYz_sg
17. Zhang WA, Yu L, Feng G. Stabilization of linear discrete-time networked control systems via protocol and controller co-design. *Int J Robust Nonlinear Control*. 2015;25(16):3072-3085.
18. Gaid MEMB, Cela A, Hamam Y. Optimal integrated control and scheduling of systems with communication constraints. Proceedings of the 44th IEEE Conference on Decision and Control; 2005:854-859.
19. Gaid MMB, Cela A, Hamam Y. Optimal integrated control and scheduling of networked control systems with communication constraints: application to a car suspension system. *IEEE Trans Control Syst Technol*. 2006;14(4):776-787.
20. Colombo A, Bahraini M, Falcone P. Measurement scheduling for control invariance in networked control systems. Proceedings of the 2018 IEEE Conference on Decision and Control (CDC); 2018:3361-3366.
21. Bahraini M, Zanon M, Falcone P, Colombo A. Robust control invariance and communication scheduling in lossy wireless networked control systems; 2020. arXiv preprint arXiv:2005.06094.
22. Bahraini M, Zanon M, Colombo A, Falcone P. Receding-horizon robust online communication scheduling for constrained networked control systems. Proceedings of the 2019 18th European Control Conference (ECC); 2019:2969-2974.
23. Bahraini M, Zanon M, Colombo A, Falcone P. Optimal control design for perturbed constrained networked control systems. *IEEE Control Syst Lett*. 2020;5(2):553-558.
24. Puterman ML. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons; 2014.
25. Scokaert PO, Mayne DQ. Min-max feedback model predictive control for constrained linear systems. *IEEE Trans Automat Contr*. 1998;43(8):1136-1142.

26. Chisci L, Rossiter JA, Zappa G. Systems with persistent disturbances: predictive control with restricted constraints. *Automatica*. 2001;37(7):1019-1028.
27. Mayne DQ, Langson W. Robustifying model predictive control of constrained linear systems. *Electron Lett*. 2001;37(23):1422-1423.
28. Kolmanovsky I, Gilbert EG. Theory and computation of disturbance invariant sets for discrete-time linear systems. *Math Probl Eng*. 1998;4(4):317-367.
29. Borrelli F, Bemporad A, Morari M. *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press; 2017.
30. Gilbert EG, Tan KT. Linear systems with state and control constraints: the theory and application of maximal output admissible sets. *IEEE Trans Automat Contr*. 1991;36(9):1008-1020.
31. Bertsekas D. Infinite time reachability of state-space regions by using feedback control. *IEEE Trans Automat Contr*. 1972;17(5):604-613.
32. Raković S, Kerrigan EC, Mayne DQ, Kouramas KI. Optimized robust control invariance for linear discrete-time systems: theoretical foundations. *Automatica*. 2007;43(5):831-841.
33. Bar-Noy A, Ladner RE. Windows scheduling problems for broadcast systems. *SIAM J Comput*. 2003;32(4):1091-1113.
34. Fishburn PC, Lagarias JC. Pinwheel scheduling: achievable densities. *Algorithmica*. 2002;34(1):14-38.
35. Bar-Noy A, Ladner RE, Tamir T. Windows scheduling as a restricted version of bin packing. *ACM Trans Algorithms (TALG)*. 2007;3(3):224-233.

How to cite this article: Bahraini M, Zanon M, Colombo A, Falcone P. Optimal scheduling and control for constrained multi-agent networked control systems. *Optim Control Appl Meth*. 2021;1-21. <https://doi.org/10.1002/oca.2777>