Co-design of Polynomial Control Law and Communication Scheduling Strategy for Multi-Loop Networked Control Systems

Anusree Rajan, Ajay Kattepur, and Pavankumar Tallapragada

Abstract— **This paper deals with the problem of stabilization of a multi-loop networked control system over a shared communication network. We co-design a polynomial control law and a communication scheduling strategy, which is based on the earliest deadline first (EDF) algorithm. The proposed polynomial control strategy allows for generating a time-varying control input to the plant of each control subsystem, even between two successive communication instants, by transmitting limited information over the communication network and by using limited computational resources at the actuator. We provide a sufficient condition that ensures that the proposed communication scheduling strategy guarantees global asymptotic stability of the origin of each control subsystem. We illustrate our results through numerical examples.**

I. INTRODUCTION

Recent advances in communication technology have increased the popularity of multi-loop networked control systems in different fields of applications such as traffic control of mobile robots in factories, formation control of a swarm of robots, and real-time control of autonomous vehicles. In many of these applications, the communication network is shared and limited [1]. Thus, it is important to ensure efficient utilization of communication resources while achieving the control objective. With this motivation, we co-design a control and communication scheduling strategy for stabilization of multi-loop networked control systems over shared communication networks.

Literature Review: There are many papers such as [2]–[4] on communication resource scheduling methods with a focus on optimizing the performance of the network such as maximizing throughput or minimizing communication latency. But these papers typically ignore the requirements of control tasks over networks. In the networked control systems literature, reference [5] computes the minimum-variance control performance under various common medium access control (MAC) protocols using numerical methods. Reference [6] considers the state estimation problem for multiple plants over a shared communication network and compares the performance of MAC protocols in terms of the communication frequency and the estimation error covariance. On the other hand, [7], [8] propose event-triggered or self-triggered control methods over a shared communication network.

The tight coupling between control and communication in networked control systems necessitates a co-design of the two, such as in [9]. An important factor while co-designing the control and communication strategies is how the actuator generates the control input to the plant when there is no feedback about the plant's state. Reference [10] considers the control input to the plant as zero when there is no communication from the controller to the actuator. This paper provides sufficient conditions for the existence of a scheduling strategy that ensures exponential stability of each control subsystem. Reference [11] deals with the problem of co-designing an optimal control law and an event-triggered scheduling law that minimizes an average cost criterion for stochastic multi-loop networked control systems. In this problem setup, when there is no communication, the control input to the plant is generated by using a model-based state estimator at the actuator. Reference [12] co-designs an event based control and scheduling strategy by solving an optimization problem with a quadratic cost function. In this approach, control input to each plant is held constant between two successive communication instants. Reference [13] proposes a self-triggered model-predictive control (MPC) method for network scheduling and control of a multi-loop system. In this method, for each control loop, at each sampling instant, a centralized controller computes the optimal piecewise constant control signal as well as the optimal time to wait before taking the next sample. Reference [14] also considers an MPC based multi-loop control system. This paper proposes an algorithm to co-design MPC and communication scheduling protocol to ensure stability of each control loop under a restrictive assumption on the number of control loops.

Contributions: The contributions of this paper are:

- We co-design a polynomial control law and a communication scheduling strategy, based on the earliest deadline first algorithm, for stabilization of multi-loop networked control systems over shared communication networks. We provide a sufficient condition that guarantees global asymptotic stability of each control subsystem.
- Compared to the model-based control method, the proposed polynomial control method requires less computational resources at the actuator and also provides greater privacy and security.
- Compared to the MPC-based control method, at each event, our proposed method requires only a limited number of parameters to be sent irrespective of the time duration of the signal.

Notation: Let $\mathbb{R}$ denote the set of all real numbers. Let $\mathbb{Z}$, $\mathbb{N}$ and $\mathbb{N}_0$ denote the set of all integers, positive and non-negative integers, respectively. For $a, b \in \mathbb{R}$, we let $[a,b]_{\mathbb{Z}} := [a,b] \cap \mathbb{Z}$ and $[a,b)_{\mathbb{Z}} := [a,b) \cap \mathbb{Z}$. For any $x \in \mathbb{R}^n$, $\|x\|$ denotes the euclidean norm. For a square matrix $A \in \mathbb{R}^{n \times n}$ with real eigenvalues, let $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ denote the smallest

Anusree Rajan was a research student intern at Ericsson Research, Bengaluru. Ajay Kattepur is with Ericsson Research, Bengaluru and Pavankumar Tallapragada is with the Indian Institute of Science, Bengaluru. anusreerajan@iisc.ac.in, ajay.kattepur@ericsson.com, pavant@iisc.ac.in

and the largest eigenvalues of A, respectively. Further, for a symmetric matrix $A \in \mathbb{R}^{n\times n}$, $A \succeq 0$ and $A \prec 0$ mean that A is positive semi-definite and negative definite, respectively.

II. Problem Setup

In this section, we present the system dynamics, form of the polynomial control law, model of the communication network and communication scheduler. Finally, we present the objective of this paper.

A. System Dynamics

Consider the multi-loop networked control system shown in Figure 1 with N physically decoupled control subsystems

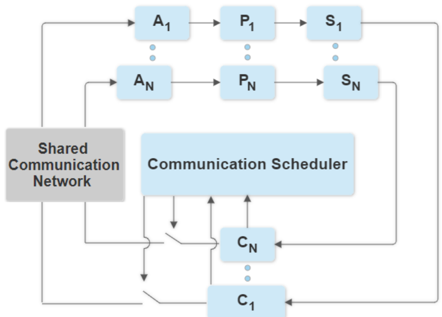


Fig. 1: Multi-loop networked control system.

whose feedback loops are closed over a shared communication network. Control subsystem i, for $i \in [1,N]_{\mathbb{Z}}$, consists of a plant $\mathbf{P}_i$, a sensor $\mathbf{S}_i$ a remote controller $\mathbf{C}_i$ and an actuator $\mathbf{A}_i$. Plant $\mathbf{P}_i$, for $i \in [1,N]_{\mathbb{Z}}$, has the following dynamics,

$$x_i(t+1) = A_i x_i(t) + B_i u_i(t), \quad \forall t \in \mathbb{N}_0, \tag{1}$$

where $x_i(t) \in \mathbb{R}^{n_i}$ denotes the state of the plant and $u_i(t) \in \mathbb{R}^{m_i}$ denotes the control input to the plant. A_i and B_i are system matrices of $\mathbf{P}_i$ with appropriate dimensions.

B. Polynomial Control Law

The remote controller of each subsystem has access to the state information of the corresponding plant at every time instant. Whenever allowed by the communication scheduler, the controller transmits a data packet to the actuator over the shared communication network. The actuator generates the control input to the plant based on the data packets received from the controller. We let $\left(t_k^{\{i\}}\right)_{k\in\mathbb{N}_0}$ denote the sequence of communication time instants from the controller $\mathbf{C}_i$ to the actuator $\mathbf{A}_i$, for all $i \in [1,N]_{\mathbb{Z}}$. Further, $\left(r_k^{\{i\}}\right)_{k\in\mathbb{N}_0}$ is the sequence of time instants at which the actuator $\mathbf{A}_i$ updates the control signal. At $r_k^{\{i\}}$, the actuator $\mathbf{A}_i$ updates the control signal based on the data that the controller sends at $t_k^{\{i\}}$.

In order to apply a time varying control input to the plant, even between two successive update instants, while also transmitting limited information over the shared communication network, we consider a polymonial control law for each subsystem. That is, $\forall i \in [1,N]_{\mathbb{Z}}$, each control input to the plant $\mathbf{P}_i$ is a polynomial of degree p_i as given below.

$$u_i(r_k^{\{i\}} + \tau) = \Phi_i(\tau)\mathbf{a}_i(k), \ \forall \tau \in [0, r_{k+1}^{\{i\}} - r_k^{\{i\}})_{\mathbb{Z}}, \tag{2}$$

where

$$\Phi_i(\tau) := \begin{bmatrix} \phi_i^\top(\tau) & 0 & \dots & 0 \\ 0 & \phi_i^\top(\tau) & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & \phi_i^\top(\tau) \end{bmatrix} \in \mathbb{R}^{m_i \times m_i(p_i+1)},$$

$\phi_i^\top(\tau) := \begin{bmatrix} 1 & \tau & \dots & \tau^{p_i} \end{bmatrix}$ and $\mathbf{a}_i(k) \in \mathbb{R}^{m_i(p_i+1)}$. Note that, the k^{th} data packet transmitted from the controller to the actuator at $t_k^{\{i\}}$ contains $\{\mathbf{a}_i(k), r_k^{\{i\}}\}$, where $\mathbf{a}_i(k)$ are the new coefficients of the polynomial control signal that need to be used by the actuator from $r_k^{\{i\}}$. We assume that the data packet transmitted by the controller at $t_k^{\{i\}}$ is received by the actuator before $t_k^{\{i\}}+1$ and hence $r_k^{\{i\}}$ must satisfy

$$r_k^{\{i\}} \geq t_k^{\{i\}} + 1.$$

Note that the polynomial control law (2) is a special case of the parameterized control law proposed in [15].

C. Communication Network and Scheduler

As the communication network is shared and has limited capacity, at each time instant, only a subset of control subsystems can access the network. For simplicity, we assume that only one subsystem can access the channel at a time. Here, we consider a centralized communication scheduler that dynamically schedules access for the communication network to the set of control subsystems. Whenever a control subsystem i needs access to the communication network, the controller $\mathbf{C}_i$ sends a priority signal to the communication scheduler. The communication scheduler provides network access to the control subsystems based on such priority signals. When a control subsystem i gets access to the communication network, the controller $\mathbf{C}_i$ generates the data packet and communicates the same to the actuator $\mathbf{A}_i$.

D. Objective

Here our objective is to co-design the polynomial control law and the communication scheduling strategy so that each control subsystem is globally asymptotically stable. Specifically, for each control subsystem i, we want to design policies for determining the coefficients of the polynomial control law $\{\mathbf{a}_i(k)\}_{k\in\mathbb{N}_0}$ and the priority signal, which implicitly determines the sequences $(t_k^{\{i\}})_{k\in\mathbb{N}_0}$ and $(r_k^{\{i\}})_{k\in\mathbb{N}_0}$, to guarantee global asymptotic stability.

III. Co-design of the Polynomial Control Law and the Communication Scheduling Strategy

In this section, we design the polynomial control law and the communication scheduling strategy to meet the objective.

A. Design of Polynomial Control Law

Let us first see how to choose $\mathbf{a}_i(k)$, the coefficients of the polynomial control law at the communication time instant $t_k^{\{i\}}$, given the updation time instant $r_k^{\{i\}}$. In Section III-B, we specify a method for choosing $r_k^{\{i\}}$. Note that, $\forall t \in [t_k^{\{i\}}, r_k^{\{i\}})_{\mathbb{Z}}$, the actuator $\mathbf{A}_i$ generates the control input $u_i(t)$ by using the coefficients $\mathbf{a}_i(k-1)$ which are determined at $t_{k-1}^{\{i\}}$. So, at each communication time instant $t_k^{\{i\}}$, the controller $\mathbf{C}_i$ first predicts the state of the plant $\mathbf{P}_i$ at $r_k^{\{i\}}$, which we denote as $x_i(r_k^{\{i\}}|t_k^{\{i\}})$. Then, it determines the new coefficients $\mathbf{a}_i(k)$ by solving the following finite horizon optimization problem,

$$\begin{aligned} \mathbf{a}_i(k) \in \; & \underset{a\in\mathbb{R}^{m_i(p_i+1)}}{\arg\min} \quad V_i(\hat{x}_i(r_k^{\{i\}}+L_i)), \\ & \text{s.t.} \;\; \hat{x}_i(t+1) = A_i\hat{x}_i + B_i\Phi_i(t-r_k^{\{i\}})a, \\ & \forall t \in [r_k^{\{i\}}, r_k^{\{i\}}+L_i-1]_{\mathbb{Z}}, \; \hat{x}_i(r_k^{\{i\}}) = x_i(r_k^{\{i\}}|t_k^{\{i\}}), \end{aligned} \tag{3}$$

where $V_i(x_i) := x_i^\top P_i x_i$ is a candidate Lyapunov-like function with $P_i > 0$ and $L_i \in \mathbb{N}$ is a design parameter.

Here, we assume that the controller has perfect knowledge about the dynamics of the corresponding plant. So, given the current state and the control input, it can exactly predict the future state. That is, we can say that, with $\Delta_k^{\{i\}} := r_k^{\{i\}} - r_{k-1}^{\{i\}}$,

$$x_i(r_k^{\{i\}}|t_k^{\{i\}}) = x_i(r_k^{\{i\}}) = F_i(\Delta_k^{\{i\}})x_i(r_{k-1}^{\{i\}}) + G(\Delta_k^{\{i\}})\mathbf{a}_i(k-1),$$
$$\text{where} \quad F_i(\tau) := A_i^\tau, \quad G_i(\tau) := \sum_{j=0}^{\tau-1} A_i^{\tau-1-j}B_i\Phi_i(j).$$

Similarly, the closed form expression for $\hat{x}_i(r_k^{\{i\}}+L_i)$ is

$$\hat{x}_i(r_k^{\{i\}}+L_i) = F_i(L_i)\hat{x}_i(r_k^{\{i\}}) + G_i(L_i)a.$$

Using this fact, we can rewrite the optimization problem (3) as the following unconstrained optimization problem,

$$\begin{aligned} & \mathbf{a}_i(k) \in \underset{a\in\mathbb{R}^{m_i(p_i+1)}}{\arg\min} \quad J_i(a), \\ & \text{where} \quad J_i(a) = \left(\hat{x}_i(r_k^{\{i\}}+L_i)\right)^\top P_i \left(\hat{x}_i(r_k^{\{i\}}+L_i)\right). \end{aligned} \tag{4}$$

Remark 1. *The unconstrained optimization problem (4) is a convex (and quadratic) optimization problem as the Hessian of the cost function is* $2G_i(L_i)^\top P_i G_i(L_i) \succeq 0$. •

As the optimization problem (4), which is equivalent to (3), is an unconstrained convex optimization problem, the stationarity condition is necessary and sufficient for optimality. Here, the stationarity condition $\frac{\partial J(a)}{\partial a} = 0$ implies that $M_i(L_i)a + W_i(L_i)\hat{x}_i(r_k^{\{i\}}) = 0$, where

$$M_i(L_i) := G_i^\top(L_i)P_iG_i(L_i), \quad W_i(L_i) := G_i^\top(L_i)P_iF_i(L_i).$$

Note that the optimization problem (4) may have an infinite number of solutions. So, we choose the optimizer with minimum norm. That is, we choose

$$\mathbf{a}_i(k) = -M_i^+(L_i)W_i(L_i)\hat{x}_i(r_k^{\{i\}}), \tag{5}$$

where M_i^+ denotes the pseudoinverse of M_i. Note that L_i in (4), and hence in (5), is a design parameter. In particular, we want to choose L_i so as to guarantee asymptotic stabilization of zero for the i^{th} subsystem. The following result, is a step towards that.

Proposition 2. *For $i \in [1,N]_{\mathbb{Z}}$, consider the system (1)-(2) and the Lyapunov-like function (3). Suppose for all $k \in \mathbb{N}_0$, $r_{k+1}^{\{i\}} \geq r_k^{\{i\}} + L_i$ for $L_i \in \mathbb{N}$. Suppose L_i satisfies*

$$E_i := F_i^\top(L_i)P_iF_i(L_i) - W_i^\top(L_i)M_i^+(L_i)W_i(L_i) - P_i \prec 0. \tag{6}$$

Then, $\forall k \in \mathbb{N}_0$, $V_i(x_i(r_k^{\{i\}}+L_i)) \leq \alpha_i V_i(x_i(r_k^{\{i\}}))$, for some $\alpha_i \in [0,1)$.

Proof. As the controller has perfect knowledge about the dynamics of the system, and since $r_{k+1}^{\{i\}} \geq r_k^{\{i\}} + L_i$, we have that $x_i(r_k^{\{i\}}|t_k^{\{i\}}) = x_i(r_k^{\{i\}})$, $\forall k \in \mathbb{N}_0$. Hence, $\hat{x}_i(r_k^{\{i\}}+\tau) = x_i(r_k^{\{i\}}+\tau)$, $\forall \tau \in [0, L_i]_{\mathbb{Z}}$. Then, from (5) we can say that,

$$\begin{aligned} & V_i(x_i(r_k^{\{i\}}+L_i)) = x_i^\top(r_k^{\{i\}}+L_i)P_ix_i(r_k^{\{i\}}+L_i) \\ & = [F_i(L_i)x_i(r_k^{\{i\}}) + G_i(L_i)a]^\top P_i[F_i(L_i)x_i(r_k^{\{i\}}) + G_i(L_i)a] \\ & = x_i^\top(r_k^{\{i\}})[F_i^\top(L_i)P_iF_i(L_i) - W_i^\top(L_i)M_i^+(L_i)W_i(L_i)]x_i(r_k^{\{i\}}), \end{aligned}$$

where the last equation follows from the facts that $M_i(L_i)$ and $M_i^+(L_i)$ are symmetric as well as that $M_i^+(L_i)M_i(L_i)M_i^+(L_i) = M_i^+(L_i)$. Now, if L_i satisfies 6, then we can say that $V_i(x_i(r_k^{\{i\}}+L_i)) - V_i(x_i(r_k^{\{i\}})) = x_i^\top(r_k^{\{i\}})E_ix_i(r_k^{\{i\}}) \leq \lambda_{\max}(E_i)\left\|x_i(r_k^{\{i\}})\right\|^2$. This implies that $V_i(x_i(r_k^{\{i\}}+L_i)) \leq \alpha_i V_i(x_i(r_k^{\{i\}}))$ where $\alpha_i := 1 + \frac{\lambda_{\max}(E_i)}{\lambda_{\max}(P_i)} \in [0,1)$ as $-\lambda_{\max}(P_i) \leq \lambda_{\max}(E_i) < 0$. □

Remark 3. *If the pair (A_i, B_i) is stabilizable, then there always exists an $L_i \in \mathbb{N}$ that satisfies the inequality (6) for the choice of $P_i \succ 0$ which is a solution of the Lyapunov equation $(A_i+B_iK_i)^\top P_i(A_i+B_iK_i) - P_i = -Q_i$, for some $Q_i \succ 0$ and $K_i \in \mathbb{R}^{m_i \times n_i}$ such that $A_i + B_iK_i$ is Schur stable.* •

Remark 4. *Note that, the control law based on zero-order-hold (ZOH) is a special case of the polynomial control law (2) with $p_i = 0$. Thus, for a given choice of L_i, the optimal value of the objective function in (4) is always less than or equal to the value corresponding to a ZOH control input. Thus, in general, there exist larger L_i for $p_i > 0$ compared to the case with $p_i = 0$ that satisfy (6).* •

B. Design of Communication Scheduling Strategy

In this paper, we consider the communication scheduling problem as a real-time task scheduling problem [16]. Specifically, we model it as multiple instances of N tasks to be scheduled on a single processor. Let $c_k^{\{i\}}$ and $T_k^{\{i\}}$, respectively, denote the release time and the relative deadline of the k^{th} instance of task i. Let $d_k^{\{i\}} := c_k^{\{i\}} + T_k^{\{i\}}$ denote the absolute deadline of the k^{th} instance of task i. Hence, the k^{th} instance of task i must be completed during $[c_k^{\{i\}}, d_k^{\{i\}}]_{\mathbb{Z}}$.

In Algorithm 1, we present the overall control and communication method for the i^{th} subsystem, with $i \in [1,N]_{\mathbb{Z}}$. Specifically, this algorithm iteratively provides the sequence of communication time instants $(t_k^{\{i\}})_{k\in\mathbb{N}_0}$, updation time instants $(r_k^{\{i\}})_{k\in\mathbb{N}_0}$, and coefficients of the polynomial control law $(\mathbf{a}_i(k))_{k\in\mathbb{N}_0}$. In Algorithm 1, $\forall i \in [1,N]_{\mathbb{Z}}$, we choose L_i that satisfies (6) and $\bar{L}_i > L_i$. We use the variable q_i to keep track of whether subsystem i is active or not. Subsystem is active ($q_i = 1$) if it seeks access to the shared network and inactive ($q_i = 0$) otherwise. At $c_k^{\{i\}}$, the release time of the k^{th} instance of task i, we set $q_i = 1$. At each time step t, subsystem i sends $\{q_i, d^{\{i\}}\}$ to the central scheduler, which gives access to the network for at most one of the subsystems.

Algorithm 1: Overall control and communication algorithm for subsystem $i \in [1,N]_{\mathbb{Z}}$

Input: $A_i, B_i, x_i(0), L_i, \bar{L}_i$
Output: $t_k^{\{i\}}, r_k^{\{i\}}, \mathbf{a}_i(k),\ \forall k \in \mathbb{N}_0$

1 **Initialize:** $t=0,\ k=0,\ q_i=1,\ c_0^{\{i\}}=0,\ d_0^{\{i\}}=L_i-1$
2 **while do**
3   **if** $t = c_k^{\{i\}}$ **then**
4     $q_i \leftarrow 1$
5   **end**
6   $d^{\{i\}} \leftarrow d_k^{\{i\}}$
7   **Send** $\{q_i, d^{\{i\}}\}$ to scheduler (Algorithm 2)
8   **Get** $\gamma_i(t)$ from scheduler (Algorithm 2)
9   **if** $\gamma_i(t)=1,$ **then**
10     $t_k^{\{i\}} \leftarrow t$
11     $r_k^{\{i\}} \leftarrow d_k^{\{i\}}+1,$
12   **Determine** $\mathbf{a}_i(k)$ by (5)
13   **Send** packet $\{\mathbf{a}_i(k), r_k^{\{i\}}\}$ to actuator $\mathbf{A}_i$
14     $c_{k+1}^{\{i\}} \leftarrow r_k^{\{i\}}$
15     $T_{k+1}^{\{i\}} \leftarrow \min\{\min\{L \geq L_i\ :\ V_i(x_i(r_k^{\{i\}}+L+1|t)) > \alpha_i V_i(x_i(r_k^{\{i\}}|t))\}, \bar{L}_i\}-1$
16     $d_{k+1}^{\{i\}} \leftarrow c_{k+1}^{\{i\}} + T_{k+1}^{\{i\}}$
17     $q_i \leftarrow 0$
18     $k \leftarrow k+1$
19   **end**
20   $t \leftarrow t+1$
21 **end**

In particular,

$$\gamma_i(t) := \begin{cases} 1, & \text{if subsystem } i \text{ gets access to the network at } t \\ 0, & \text{otherwise.} \end{cases}$$

After getting access to the communication network at $t_k^{\{i\}}$, the subsystem sets $r_k^{\{i\}} = d_k^{\{i\}}+1$. It then determines the coefficients $\mathbf{a}_i(k)$ by (5) and transmits the control packet $\{\mathbf{a}_i(k), r_k^{\{i\}}\}$ to the actuator $\mathbf{A}_i$. After transmitting the control packet, the subsystem changes $q_i = 0$. Then, it sets the release time of the next instance of task i as $c_{k+1}^{\{i\}} = r_k^{\{i\}}$. q_i again becomes 1 at $t = c_{k+1}^{\{i\}}$. That is, once the actuator $\mathbf{A}_i$ updates the coefficients of the polynomial control law to the latest value $\mathbf{a}_i(k)$ at $r_k^{\{i\}}$, the controller $\mathbf{C}_i$ again requests for network access and the whole process repeats. The absolute deadline $d^{\{i\}} = d_k^{\{i\}}$ acts as the priority signal to the central scheduler. We make the following assumption.

(A1) The control input to each plant until the zeroth updation time is zero. That is, $u_i(t)=0,\ \forall t \in [0, r_0^{\{i\}})_{\mathbb{Z}},\ \forall i \in [1,N]_{\mathbb{Z}}$.

Hence, we arbitrarily choose the initial deadline $d_0^{\{i\}} = T_0^{\{i\}} = L_i - 1$. For subsequent packets, we compute the relative deadline in Step 15 of the algorithm given $x_i(r_k^{\{i\}}|t)$. Note that the idea for this computation is very similar to that of the condition (6) though in (6), we compute a lower bound on the relative deadline over all possible values of $x_i(r_k^{\{i\}}|t)$.

Next, at each time step, the communication scheduler uses Algorithm 2 to dynamically allocate the network to at most one of the subsystems. In this algorithm, $\mathbb{I}(t)$ is the

Algorithm 2: Earliest deadline first (EDF) algorithm for communication scheduling

Input: $t,\ \{q_i, d^{\{i\}}\},\ \forall i \in [1,N]_{\mathbb{Z}}$
Output: $\gamma_i(t),\ \forall i \in [1,N]_{\mathbb{Z}}$

$\mathbb{I}(t) \leftarrow \{j \in [1,N]_{\mathbb{Z}} : q_j(t) = 1\}$
foreach $i \in [1,N]_{\mathbb{Z}}$ **do**
  $\gamma_i(t) \leftarrow \begin{cases} 1, & \text{if } i = \max\left\{\arg\min_{j \in \mathbb{I}(t)}\{d^{\{j\}}\}\right\} \\ 0, & \text{otherwise} \end{cases}$
end

set of subsystems that request for access to the network at time t. The scheduler allocates the network at time t to the subsystem with the earliest absolute deadline among all the subsystems requesting access. If there are multiple subsystems with the earliest deadline, the scheduler gives access to the subsystem with the highest subsystem id among those with the earliest deadline. Note that if $q_i = 0$ at time t, subsystem i may skip steps 7 and 8 in Algorithm 1 and directly set its $\gamma_i(t) = 0$. Similarly, Algorithm 2 works equally well if only those subsystems requiring network access send $\{q_i, d^{\{i\}}\}$ to the scheduler. Finally, if $\mathbb{I}(t)$ is empty then no subsystem gets access to the network on time step t.

IV. Analysis of the Multi-Loop Control System

In this section, we analyze the performance of the multi-loop networked control system under the proposed polynomial control law (2) and Algorithms 1 and 2. We first make the following observations regarding Algorithm 1.

Remark 5. *According to Algorithm 1,* $c_{k+1}^{\{i\}} = r_k^{\{i\}} = d_k^{\{i\}} + 1 = c_k^{\{i\}} + T_k^{\{i\}} + 1$ *and* $\bar{L}_i - 1 \geq T_k^{\{i\}} \geq L_i - 1,\ \forall k \in \mathbb{N}_0$ *and* $\forall i \in [1,N]_{\mathbb{Z}}$. *Thus, we can say that the inter-release time of task i,* $c_{k+1}^{\{i\}} - c_k^{\{i\}} \in [L_i, \bar{L}_i],\ \forall k \in \mathbb{N}_0$ *and* $\forall i \in [1,N]_{\mathbb{Z}}$. •

In the literature on real time task scheduling problems, a scheduling algorithm is said to be feasible if it can generate a schedule that ensures that none of the tasks misses its deadlines. Based on Remark 5, we next provide a sufficient condition for Algorithms 1 and 2 to be feasible.

Lemma 6. *(Sufficient condition for feasibility of Algorithm 1). Consider the multi-loop networked control system (1) under the polynomial control law (2) and Algorithms 1 and 2. Algorithms 1 and 2 are feasible if* $\sum_{i=1}^{N} \frac{1}{L_i} \leq 1$.

Proof. Note that we can model the communication scheduling problem considered in this paper as a sporadic task scheduling model [16], where the inter-release times of each task are lower bounded by a known constant. In our case, according to Remark 5, inter-release times of task $i \in [1,N]_{\mathbb{Z}}$ are uniformly lower bounded by L_i. According to Lemma 3.2 in [16], a sporadic task set is feasible if the corresponding synchronous periodic task set, with periods $\{L_i\}_{i\in[1,N]_{\mathbb{Z}}}$, is feasible. By synchronous periodic task set, we mean a set of tasks with periodic release times and the first instance of all the tasks are released at the same time. According to Corollary 3.1 in [16], a set of N synchronous periodic tasks is feasibly scheduled by the EDF algorithm if and only if $\sum_{i=1}^{N}\frac{\mathrm{C}_i}{T_i}\leq 1$, where C_i is the maximum time required to complete task i and T_i is the inter-release time of task i. In our case $\mathrm{C}_i=1$ and $\mathrm{T}_i=L_i$ for all $i\in[1,N]_{\mathbb{Z}}$. Thus, we can say that Algorithms 1 and 2 are feasible if $\sum_{i=1}^{N}\frac{1}{L_i}\leq 1$. $\square$

Now, we present the main result of this paper.

Theorem 7. *(Global asymptotic stability of control subsystems). Consider the multi-loop networked control system* (1) *under the polynomial control law* (2) *and Algorithms 1 and 2. Then, the origin of each control subsystem is globally asymptotically stable if* $\sum_{i=1}^{N}\frac{1}{L_i}\leq 1$.

Proof. Note that, according to Lemma 6, the communication scheduling algorithm is feasible if $\sum_{i=1}^{N}\frac{1}{L_i}\leq 1$. This implies that none of the control subsystems misses its deadlines. Now, consider an arbitrary $i\in[1,N]_{\mathbb{Z}}$ and $k\in\mathbb{N}_0$. According to Algorithm 1, we can say that

$$r_{k+1}^{\{i\}}=r_k^{\{i\}}+\min\{\min\{L\geq L_i \ : \ V_i(x_i(r_k^{\{i\}}+L+1))> \alpha_i V_i(x_i(r_k^{\{i\}}))\},\bar{L}_i\}.$$

This, along with Proposition 2, implies that $V_i(x_i(t))\leq \alpha_i V_i(x_i(r_k^{\{i\}}))$, for some $\alpha_i\in[0,1)$ $\forall t\in[r_k^{\{i\}}+L_i,r_{k+1}^{\{i\}}]_{\mathbb{Z}}$. Further, $\forall\tau\in[0,L_i]_{\mathbb{Z}}$,

$$\begin{aligned}\left\|x_i(r_k^{\{i\}}+\tau)\right\| &\leq \|F_i(\tau)\|\left\|x_i(r_k^{\{i\}})\right\|+\|G_i(\tau)\|\,\|\mathbf{a}_i(k)\|\\ &\leq \max_{0\leq\tau\leq L_i}\|F_i(\tau)\|\left\|x_i(r_k^{\{i\}})\right\|+\max_{0\leq\tau\leq L_i}\|G_i(\tau)\|\,\|\mathbf{a}_i(k)\|\\ &\leq H_i\left\|x_i(r_k^{\{i\}})\right\|, \end{aligned}\tag{7}$$

for some finite $H_i>0$. Here, the last inequality follows from (5). Note that, $\forall k\in\mathbb{N}_0$, $r_{k+1}^{\{i\}}-r_k^{\{i\}}\leq\bar{L}_i$ and $V_i(r_{k+1}^{\{i\}})\leq\alpha_i V_i(r_k^{\{i\}})$ for some $\alpha_i\in[0,1)$. Thus, the sequence $(V_i(r_k^{\{i\}}))_{k\in\mathbb{N}_0}$ is an infinite lengthed sequence and is upper-bounded by a geometric sequence that converges to zero. Since V is lower bounded by zero, we can say that the sequence $(V_i(r_k^{\{i\}}))_{k\in\mathbb{N}_0}$ converges to zero. Now, the fact that $\lambda_{\min}(P_i)\|x_i\|^2\leq V_i(x_i)\leq\lambda_{\max}(P_i)\|x_i\|^2$ implies

$$\|x_i(t)\|\leq\max\{1,H_i\}\sqrt{\frac{\lambda_{\max}(P_i)}{\lambda_{\min}(P_i)}}\left\|x_i(r_0^{\{i\}})\right\|,\ \forall t\geq r_0^{\{i\}}.$$

Note also that, $\forall t\in[0,r_0^{\{i\}}]_{\mathbb{Z}}$, $\|x_i(t)\|\leq\left\|A_i^{L_i}\right\|\|x_i(0)\|$ as $u_i(t)=0$ $\forall t\in[0,r_0^{\{i\}})_{\mathbb{Z}}$. Thus, given $x_i(0)$, $\|x_i(t)\|$ is uniformly upper bounded for all $t\in\mathbb{N}_0$. Further, given the bound (7), we can say that the origin of each control subsystem is globally asymptotically stable. $\square$

V. Numerical Examples

In this section, we illustrate our results through numerical simulations.

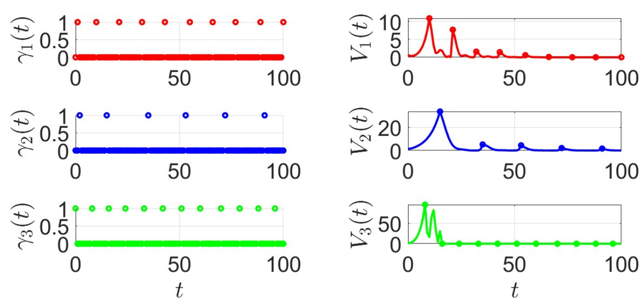
Example 1: We consider three independent control subsystems with the following dynamics,

$$x_1(t+1)=\begin{bmatrix}0.7 & -0.1 & -0.1\\ 0 & 0.8 & -0.4\\ 0 & 0 & 1.2\end{bmatrix}x_1+\begin{bmatrix}0\\0\\1\end{bmatrix}u_1,\quad\forall t\in\mathbb{N}_0,$$

$$x_2(t+1)=\begin{bmatrix}0.65 & -0.15 & -0.15\\ 0 & 0.8 & -0.3\\ 0 & 0 & 1.1\end{bmatrix}x_2+\begin{bmatrix}0\\0\\1\end{bmatrix}u_2,\quad\forall t\in\mathbb{N}_0,$$

$$x_3(t+1)=\begin{bmatrix}0.8 & -0.1 & -0.1\\ 0 & 0.9 & -0.3\\ 0 & 0 & 1.2\end{bmatrix}x_3+\begin{bmatrix}0\\0\\1\end{bmatrix}u_3.\quad\forall t\in\mathbb{N}_0,$$

We choose the design parameters $L_1=10$, $L_2=15$ and $L_3=8$ which satisfy the coditions given in Proposition 2. We choose $\bar{L}_i=L_i+5$ and the quadratic Lyapunov function $V_i(x_i):=x_i^\top P_i x_i$, where $P_i\succ 0$ is chosen such that it satisfies the Lyapunov equation $(A_i+B_iK_i)^\top P_i(A_i+B_iK_i)-P_i=-Q$, $\forall i\in[1,3]_{\mathbb{Z}}$, with Q being a 3×3 identity matrix, $K_1=\begin{bmatrix}0 & 0 & -0.3\end{bmatrix}$, $K_2=\begin{bmatrix}0 & 0 & -0.2\end{bmatrix}$ and $K_3=\begin{bmatrix}0 & -0.0333 & -0.45\end{bmatrix}$. According to Lemma 6, Algorithms 1 and 2 are feasible as $\sum_{i=1}^{3}\frac{1}{L_i}\leq 1$. We choose an arbitrary initial condition $x_1(0)=\begin{bmatrix}0.2 & 0.5 & 0.1\end{bmatrix}^\top$, $x_2(0)=\begin{bmatrix}0.5 & 0.1 & 0.3\end{bmatrix}^\top$ and $x_3(0)=\begin{bmatrix}0.1 & 0.3 & 0.4\end{bmatrix}^\top$. Here, we design each control input to each subsystem as a polynomial of degree 3, that is $p_i=3$, $\forall i\in[1,3]_{\mathbb{Z}}$. Figure 2 presents the simulation results of Example 1. Figure 2a shows the values of the indicator variable $\gamma_i(t),\forall i\in[1,3]_{\mathbb{Z}}$, where $\gamma_i(t)=1$ indicates that the control subsystem i access the network at t. Figure 2b presents the time evolution of the Lyapunov function of each control subsystem. The values of the Lyapunov functions corresponding to the updation instants $\{r_k^{\{i\}}\}$, $\forall i\in[1,3]_{\mathbb{Z}}$, $\forall k\in\mathbb{N}_0$ are presented by dots in Figure 2b. In Figure 2b, we can see that the sequence $(V_i(r_k^{\{i\}}))$ is a monotonically decreasing sequence and between two successive updation instants, the Lyapunov function may increase but it remains within a constant multiple of the norm of the state at the last update time. Thus, the origin of each control subsystem is globally asymptotically stable.



(a) Communication instants (b) Evolution of $V(x)$

Fig. 2: Simulation results of Example 1

Example 2: Next, we consider a set of 10 independent subsystems with the following dynamics, $\forall l \in [1,10]_{\mathbb{Z}}$,

$$x_l(t+1) = \begin{bmatrix} 0.7 & 0.1l & -0.1l \\ 0 & 0.8 & 0.1l \\ 0 & 0 & 1+0.1l \end{bmatrix} x_l + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u_l, \quad \forall t \in \mathbb{N}_0.$$

Note that all these subsystems are unstable when the control input is zero. We set $p_i = 3,\ \forall i \in [1,10]_{\mathbb{Z}}$. We choose the design parameters L_i as shown in Table I which satisfy the condition (6). We choose the quadratic Lyapunov function $V_i(x_i) := x_i^\top P_i x_i,\ \forall i \in [1,10]_{\mathbb{Z}}$, where $P_i > 0$ is chosen in a similar way as in Example 1. Note that, subsystems with higher degree of instability under no control input have smaller values for L_i. That is, as the degree of instability of the uncontrolled system increases, more frequent communications are required to make the system globally asymptotically stable. Note also that, according to Lemma 6, Algorithms 1 and 2 are feasible as $\sum_{i=1}^{10} \frac{1}{L_i} \leq 1$.

We consider 100 random initial conditions uniformly sampled from the unit sphere for each control subsystem and simulate the state trajectories under the proposed polynomial control law (2) and Algorithms 1 and 2. We observe that the origin of each closed loop control subsystem is globally asymptotically stable and we report the values of certain performance metrics in Table I. Here $t_c^{\{i\}}$ denotes the time taken by the value of the Lyapunov function V_i to converge to the specific value of $V_i(0) \times 10^{-4}$. $N_c^{\{i\}}$ and $N_r^{\{i\}}$, respectively, denote the number of communication and updation instants upto $t_c^{\{i\}}$. Now, let $c^{\{i\}}$ denote the ratio of the total number of communication instants to the total simulation time. Similarly, $r^{\{i\}}$ denote the ratio of the total number of updation instants to the total simulation time. Then we take the average over the set of initial conditions and these values are given in Table I. Note that control subsystems with larger values of L_i generally require more convergence time. This implies that the choice of L_i creates a tradeoff between the number of communication instants and the convergence time.

TABLE I: Numerical results of Example 2

i	L_i	$t_{c,avg}^{\{i\}}$	$N_{c,avg}^{\{i\}}$	$N_{r,avg}^{\{i\}}$	$c_{avg}^{\{i\}}$	$r_{avg}^{\{i\}}$
1	37	241.55	5.53	4.60	0.0260	0.0260
2	22	127.13	4.28	3.32	0.0430	0.0420
3	16	98.27	4.77	4.01	0.0569	0.0569
4	13	54.68	3.35	2.66	0.0729	0.0729
5	11	39.85	2.80	2.26	0.0859	0.0849
6	9	21.28	1.43	1.11	0.1049	0.1039
7	8	18.40	1.57	1.05	0.1189	0.1179
8	8	22.03	2.10	1.30	0.1179	0.1169
9	7	18.03	2.08	1.14	0.1319	0.1319
10	6	14.78	2	1.01	0.1489	0.1479

VI. Conclusion

In this paper, we considered the problem of stabilization of a multi-loop networked control system over a shared communication network. We co-designed a polynomial control law and a communication scheduling strategy, based on the earliest deadline first algorithm. We provided a sufficient condition that ensures the feasibility of the proposed communication scheduling strategy and global asymptotic stability of each control subsystem. We illustrated our results through numerical examples. Future work includes generalization of the proposed method to the case where at each time instant more than one control subsystem is allowed to access the communication network. Another potential research direction is the analysis of the robustness of the proposed control and communication strategy to disturbances due to model uncertainty and imperfections in the communication network, such as communication delay and packet drop.

References

[1] 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, 2020.

[2] S. Lu, V. Bharghavan, and R. Srikant, "Fair scheduling in wireless packet networks," *IEEE/ACM Transactions on Networking*, vol. 7, no. 4, pp. 473–489, 1999.

[3] L. Lei, Y. Kuang, X. S. Shen, K. Yang, J. Qiao, and Z. Zhong, "Optimal reliability in energy harvesting industrial wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 8, pp. 5399–5413, 2016.

[4] G. Chen, X. Cao, L. Liu, C. Sun, and Y. Cheng, "Joint scheduling and channel allocation for end-to-end delay minimization in industrial wirelesshart networks," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2829–2842, 2019.

[5] A. Cervin and T. Henningsson, "Scheduling of event-triggered controllers on a shared network," in *2008 47th IEEE Conference on Decision and Control*, 2008, pp. 3601–3606.

[6] M. Xia, V. Gupta, and P. J. Antsaklis, "Networked state estimation over a shared communication medium," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1729–1741, 2017.

[7] A. Adaldo, D. Liuzza, D. V. Dimarogonas, and K. H. Johansson, "Control of multi-agent systems with event-triggered cloud access," in *2015 European Control Conference (ECC)*, 2015, pp. 954–961.

[8] S. L. Bowman, C. Nowzari, and G. J. Pappas, "Consensus of multiagent systems via asynchronous cloud communication," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 2, pp. 627–637, 2020.

[9] Z. Lu and G. Guo, "Control and communication scheduling co-design for networked control systems: a survey," *International Journal of Systems Science*, vol. 54, no. 1, pp. 189–203, 2023.

[10] S.-L. Dai, H. Lin, and S. S. Ge, "Scheduling-and-control codesign for a collection of networked control systems with uncertain delays," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 1, pp. 66–78, 2010.

[11] A. Molin and S. Hirche, "Price-based adaptive scheduling in multi-loop control systems with resource constraints," *IEEE Transactions on Automatic Control*, vol. 59, no. 12, pp. 3282–3295, 2014.

[12] S. Al-Areqi, D. Görges, and S. Liu, "Event-based networked control and scheduling codesign with guaranteed performance," *Automatica*, vol. 57, pp. 128–134, 2015.

[13] E. Henriksson, D. E. Quevedo, E. G. W. Peters, H. Sandberg, and K. H. Johansson, "Multiple-loop self-triggered model predictive control for network scheduling and control," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 6, pp. 2167–2181, 2015.

[14] K. Liu, A. Ma, Y. Xia, Z. Sun, and K. H. Johansson, "Network scheduling and control co-design for multi-loop mpc," *IEEE Transactions on Automatic Control*, vol. 64, no. 12, pp. 5238–5245, 2019.

[15] A. Rajan and P. Tallapragada, "Event-triggered parameterized control of nonlinear systems," *IEEE Control Systems Letters*, pp. 1–1, 2024.

[16] J. Stankovic, M. Spuri, K. Ramamritham, and G. Buttazzo, *Deadline Scheduling for Real-Time Systems: EDF and Related Algorithms*, ser. The Springer International Series in Engineering and Computer Science. Springer US, 2012. [Online]. Available: https://books.google.co.in/books?id=1TrSBwAAQBAJ