A Decentralized Consistent Policy for Event-triggered Control over a Shared Contention-based Network

M. Hadi Balaghi I.[1], Duarte J. Antunes[1], Mohammad H. Mamduhi[2,3], Sandra Hirche[2]

***Abstract*— We consider a network of several independent linear systems controlled over a shared communication network. Data transmissions pertaining to each control loop are arbitrated by a scheduler collocated with the plant's sensors that transmits the state information to the corresponding remote controller collocated with the plant's actuators. The shared communication channel is assumed to be operating based on a contention-based protocol, endowing the networked control system with desirable reconfigurable and scalable features. We propose a class of scheduling policies which admit a decentralized optimal control implementation and an event-triggered policy within this class which is shown to be consistent, i.e. it results in a better control performance for any linear system, measured by an average quadratic cost than its non-event-based counterpart.**

I. Introduction

Much research has been carried out in recent years on event-triggered control [1]–[7]. The goal of event-triggered control is to reduce the communication burden and the energy expenditure of wireless sensors in networked control systems, in which spatially distributed agents belonging to one or more control loops (sensors, actuators, etc.) exchange information over a limited-resource communication network. This is achieved by jointly designing the communication protocols and control policies, as opposed to traditional control where the communication protocol, typically corresponding to periodic transmission, is fixed. The majority of the existing results consider an ideal network (no drops/collisions) available at every time-step and a single control loop, often in the framework of periodic event-triggered control (PETC). In PETC, the state or an output of the system is periodically observed and a scheduler decides whether to transmit or not this data to other control agents [8].

Significantly less research has focused on the case where several control loops are closed via a shared communication network [2], [6]. For concreteness, consider the setting depicted in Fig. 1 which will be considered throughout the

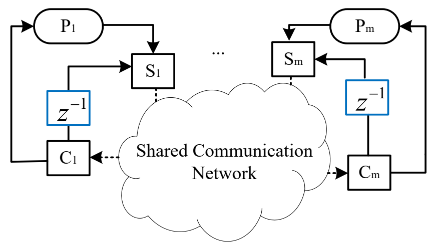

Fig. 1: Decentralized control loops over the shared communication network

paper. Each of the m independent control loops incorporates a scheduler collocated with the plant's sensors that transmits the state information to a corresponding remote controller collocated with the plant's actuators. We assume time-synchronization of the sampling process among control loops and that all have the same sampling time. However, only one scheduler can successfully transmit its data at every sampling time through the shared communication network to its corresponding controller.

The shared communication network in this setting can either operate based on contention-free protocols, such as time-division multiple access (TDMA) protocols [9], or contention-based protocols such as slotted-ALOHA [9]. Contention-free protocols would guarantee a fixed transmission rate for every user and each control loop could be designed using PETC *independently* of the other loops [10]–[12]. However, contention-based protocols offer high bandwidth when only one or a few users are actively transmitting through the network, as opposed to TDMA in which the bandwidth is partitioned a priori between all the possible network users, that are not necessarily active all the time [13]. Moreover, contention-based networks can adapt decentrally to the changes in the number of active users, and are, therefore scalable and easily reconfigurable, whereas TDMA networks need a central coordinator for the communication resource rescheduling. These properties are especially appealing in the context of the Internet of Things (IoT) [14], plug and play control [15], and decentralized control settings, and therefore we consider a contention-based protocol.

However, the design of scheduling and control strategies over a contention-based protocol is in general challenging since when multiple agents transmit, collisions can occur, i.e., data is lost when more than one users attempt to transmit concurrently. In general, if the control loops transmit independently based on their individual state-dependent events,

[1]M. Hadi Balaghi I. and Duarte J. Antunes are with the Control Systems Technology Group, Department of Mechanical Engineering, Eindhoven University of Technology, the Netherlands {m.balaghiinaloo, d.antunes}@tue.nl.

[2]Mohammad H. Mamduhi and Sandra Hirche are with the Chair of Information-oriented Control, Technical University of Munich, Germany {mh.mamduhi, hirche}@tum.de.

[3]Mohammad H. Mamduhi is also with the Department of Automatic Control, KTH Royal Institute of Technology, Stockholm, Sweden mamduhi@kth.se.

This project was funded from the European Union's Horizon 2020 Framework Programme for Research and Innovation under grant agreement No. 674875 and DFG Priority Program SPP1914 "Cyber-physical Networking".

collisions also occur in a state-dependent fashion [16], [17] and the event-triggered control (ETC) of different control loops cannot be designed independently which does not lead to a scalable design.

Therefore, in the present work, we introduce a class of scheduling policies specifying that every scheduler using the shared contention-based communication network transmits randomly with a fixed probability at every time-step. This policy enables to decouple the design of the scheduler-controller pair associated with each loop from the other control loops. Then, we propose an event-triggered scheduling policy in the class of random scheduling policies with fixed transmission probability. This policy is inspired by the one proposed in the context of PETC by which transmissions are triggered when state errors exceed stochastic thresholds [18], resulting in a random event-based scheduling policy with an adjustable transmission probability at every time-step. We refer to this policy as the stochastic threshold event-triggered transmission (STETT). As shown in [18], STETT can outperform periodic control for any linear system, when performance is measured by an average quadratic cost, which is one of the desired consistency properties for ETCs defined in [19]. As the main contribution of this work, we show that the proposed event-based scheduling policy is consistent in the sense that outperforms its non-event-based counterpart. This non-event-based scheduling policy is characterized by the scheduler transmitting purely stochastically with a constant probability at every time-step.

The remainder of the paper is organized as follows. In Section II we provide the problem setting, the assumptions on the communication network and the class of schedulers we will be considering. In Section III we find the optimal closed-loop controller when the scheduler is operating based on the PST policy. In Section IV we propose our novel scheduling policy. The consistency of the proposed policy is addressed in Section V. In Section VI we validate our results by providing numerical simulation and we provide some concluding remarks in Section VII.

Notation: $f(x|y)$ indicates the conditional probability density function (pdf) of a random variable x given the set of information y and $\mathcal{N}(\bar{y}, Y)$ indicates a Gaussian pdf with mean $\bar{y}$ and covariance Y. $\Pr(.)$ denotes the probability of an event; $\delta \sim \mathrm{B}(p)$ indicates that the random variable δ follows a Bernoulli distribution with probability p. Let $\varrho(A)$ denote the spectral radius of the square matrix A. Moreover, $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$ in which $\mathbb{N}$ is the set of natural numbers and $Z_k = \{0, 1, \ldots, k\}$.

II. Problem setting

Consider a networked control system consisting of several independent dynamic users, each of them is modeled by a linear time-invariant system as follows

$$x^i_{k+1} = A^i x^i_k + B^i u^i_k + w^i_k, \tag{1}$$

where $x^i_k \in \mathbb{R}^{n^i}$, $u^i_k \in \mathbb{R}^{n^i_u}$ are, respectively, the state and the control input at time $k \in \mathbb{N}_0$, and $(w^i_0, w^i_1, \ldots)$ is a sequence of i.i.d. Gaussian random variables with zero mean and covariance $W^i = \mathbb{E}[w^i_k w^{i\intercal}_k]$ for every $k \in \mathbb{N}_0$ and $i \in \{1, \ldots, m\}$ where m is the total number of dynamic users. A scheduler is collocated with the sensors of every system and transmits the corresponding state information to a remote controller, collocated with the actuators, over a network shared between all control loops. All the pairs (A^i, B^i) are assumed to be controllable. Moreover, control performance of each user is measured by the following average quadratic cost

$$J^i = \limsup_{T\to\infty} \frac{1}{T}\mathbb{E}[\sum_{k=0}^{T-1} x^{i\intercal}_k Q^i x^i_k + u^{i\intercal}_k R^i u^i_k] \tag{2}$$

in which Q^i is a positive semi-definite matrix and R^i is a positive definite matrix with appropriate dimensions. The pairs (A^i, Q^i) are assumed to be observable. Therefore, each dynamic user is characterized by the tuple $(A^i, B^i, W^i, Q^i, R^i)$, which in general and typically are different from the other users, i.e. the users are heterogeneous. The overall performance of the system is defined as a social cost, i.e. the sum of the performance of each individual control loop $\sum_{i=1}^m J^i$.

A. Assumptions of the communication channel

The assumptions on the communication channel are very similar to the ones considered in the context of contention-based protocols such as slotted-ALOHA [9] and can be summarized as follows:

- Time is partitioned into fixed-size slots and during each time-slot only one successful transmission is possible.
- All users are restricted to start transmitting at the beginning of the slot.
- If more than one user attempt to transmit, then there is a collision and the corresponding information is lost.
- Every user waits for a data receipt acknowledgment after an attempt for the data transmission and if an acknowledgment is not received, it is assumed that a collision has occurred. Data acknowledgments take place within the corresponding transmission time-slot.

Besides these assumptions, we make two additional assumptions suited for control applications:

- There is no mechanism for retransmission of lost data.
- The transmission time is assumed to be negligible with respect to the duration of the time-slot, i.e., the controller is assumed to access data at the beginning of the time-slot when a successful transmission occurs.

The first of these two assumptions is different from the retransmission mechanism of many standard contention-based protocols and is motivated by the fact that retransmissions in control applications would result in long delays and buffering new data instead of transmitting it immediately. Therefore, we consider that after a collision/data drop, the newest sensor information can be obtained and a new transmission with new data can be attempted. As we shall see shortly, the schedulers will be restricted such that this new transmission occurs with a constant probability p, as in the retransmission mechanism of slotted-ALOHA, except that new data is sent. Moreover, implicit in the second assumption is the fact that we consider the time-step of the discrete-time system x_k to coincide with the time-slot index, i.e., x_k pertains to the state of the system at the beginning of the time-slot.

B. Admissible class of schedulers

In this section, we introduce a class of scheduling policies that leads to a decentralized structure for the optimal controller design for the users of the shared contention-based communication network. Let

$$\delta_k^i = \begin{cases} 1, & \text{if the scheduler } i \text{ attempts data transmission} \\ 0, & \text{otherwise} \end{cases}$$

and

$$\rho_k^i = \begin{cases} 1, & \text{if the network is available for the user } i \\ 0, & \text{otherwise} \end{cases}$$

for every user $i \in \{1, \ldots, m\}$ at every time-step $k \in \mathbb{N}_0$. Based on the properties of the shared contention-based communication network introduced in Section II-A, one can conclude that ρ_k^i is given by

$$\rho_k^i = \prod_{j=1, j\neq i}^{m} (1 - \delta_k^j).$$

Moreover, consider

$$\sigma_k^i = \rho_k^i \delta_k^i$$

as a variable indicating a successful transmission at every time-step, in which case $\sigma_k^i = 1$, and $\sigma_k^i = 0$, otherwise. In the following, we introduce a class of admissible triggering policies specifically introduced for providing a decentralized control design structure when the communication network is contention-based.

Definition 1: The class of admissible scheduling policies for the control loops using the proposed shared contention-based communication network in Section II-A is characterized by

$$\Pr(\delta_k^i = 1) = p^i, \tag{3}$$

where $p^i \in [0, 1]$ is a constant transmission probability for every $i \in \{1, \ldots, m\}$ and at every time-step $k \in \mathbb{N}_0$.

If the scheduler of all control loops comply with (3), then from a single control loop perspective, the contention-based communication network can be abstracted as if at each time-step there is a constant probability

$$q^i = \prod_{j=1, j\neq i}^{m} (1 - p^i) \tag{4}$$

that all the other users are not trying to transmit and the network is available. Note that this abstraction is irrespective of the state vector of the other dynamic users. Therefore, at every time-step, the control loop of interest has a successfully transmitting probability of $q^i p^i$ which directly affects the stability and the performance of that control loop. This will be thoroughly discussed in the following sections.

Assumption 1: The schedulers of all control loops using the shared contention-based communication network are operating based on a policy within the class of admissible scheduling policies (3) where all the values of p^i for $i \in \{1, \ldots, m\}$ are given a priori.

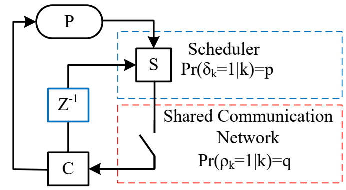

Fig. 2: A decoupled control loop of the shared contention-based communication network

Therefore, by assuming given values for p^i when $i \in \{1, \ldots, m\}$, we can drop the index i and analyze every closed-loop control system independently from the other control loops, as depicted in Fig. 2. The following remark provides a rule of thumb for optimally selecting the triggering probability of the schedulers when all of them are forced to transmit with equal probabilities [9].

Remark 1: Assume a constant value is aimed to assign to the triggering probability of all the schedulers when they are operating based on the proposed admissible scheduling policy (3), i.e. $p^i = p^*$ for all $i \in \{1, \ldots, m\}$. Then it can be shown that the proposed contention-based network has its maximum throughput value at every time-step if

$$p^* = \frac{1}{m}.$$

C. Problem statement

Note that based on the structure of the shared communication network in section II-A, we assume that the scheduler can receive an error-free acknowledgment signal from the controller whenever an attempted transmission is successful, and therefore it knows all the previous values of ρ_k. Accordingly, in order to decide on δ_k the scheduler has the following information set

$$\mathcal{I}_k = \{\delta_\ell, x_\ell | \ell \in Z_{k-1}\} \cup \{\rho_\ell | \delta_\ell = 1 \wedge \ell \in Z_{k-1}\} \cup \{x_k\}. \tag{5}$$

If the scheduler decides to use this information, we call it an event-triggered or even-based scheduler. Otherwise, i.e., if the scheduler triggers in a non-event-based fashion while it complies with (3), it should transmit purely stochastically as defined next:

Definition 2: An admissible non-event-based scheduling policy is to trigger data purely stochastically with a constant probability at every time-step, i.e.

$$\delta_k^{ps} \sim \mathrm{B}(p). \tag{6}$$

We call this scheduling policy as purely stochastic transmission (PST) policy. □

As we will show in the sequel (see Theorem 1), given this scheduling policy one can compute the optimal control law. Now inspired by [19], let us define the following consistency properties:

Definition 3: An event-triggered control policy, consisting of a scheduler and a control policy, is called consistent if

1) It results in a better performance, measured by the average quadratic cost (2), than that of the control loop when the scheduler follows a purely stochastic control policy and the controller is the corresponding optimal controller.
2) When there is zero disturbance input, the scheduler stops data triggering.

Then the problem we are interested in can be stated as follows: Find a consistent admissible event-based scheduling and control policy for each control loop.

III. Purely stochastic transmission policy

In this section, we find the associated optimal control law when the scheduler is operating based on the PST policy. In fact, when the scheduler of a control loop follows the PST policy (6), the controller is proved to be linear and an analytical closed-form expression for the control performance is provided in Theorem 1. We need the following assumption for the performance index (2) to be bounded which holds when A is stable or qp is sufficiently large [20].

Assumption 2: $\varrho(\sqrt{1-qp}A) < 1$.

Theorem 1: Suppose that Assumptions 1 and 2 hold. Then, when the scheduler transmits based on the PST policy (6) and $\rho_k \sim \mathrm{B}(q)$ for $q \in [0, 1]$, the optimal control policy minimizing the average quadratic cost is

$$u_k = K\hat{x}_{k|k} \tag{7}$$

where

$$\begin{aligned} \hat{x}_{k+1|k} &= A\hat{x}_{k|k} + Bu_k, \\ \hat{x}_{k|k} &= \begin{cases} x_k, & \text{if } \sigma_k = 1 \\ \hat{x}_{k|k-1}, & \text{otherwise.} \end{cases} \end{aligned} \tag{8}$$

and

$$\begin{aligned} K &= -(B^\mathsf{T}PB + R)^{-1}B^\mathsf{T}PA, \\ P &= A^\mathsf{T}PA + Q - K^\mathsf{T}(B^\mathsf{T}PB + R)K. \end{aligned}$$

Moreover, this control loop is mean square stable, i.e. $\sup\{\mathbb{E}[x_k x_k^\mathsf{T}]|k\in\mathbb{N}_0\} \le c$ and its corresponding average control performance is

$$J^{ps} = \mathrm{tr}(PW) + \sum_{i=0}^{\infty}(1-qp)^{i+1}\mathrm{tr}(A^iWA^{\mathsf{T}i}Y) \tag{9}$$

where $Y = K^\mathsf{T}(B^\mathsf{T}PB + R)K$.

□

The proof of Theorem 1 is omitted due to space limitation.

IV. Proposed policy using stochastic thresholds

The novel scheduling policy proposed in this work results from, first, picking the parameters of the STETT policy provided in [18] so that the probability that a transmission attempt occurs is constant, and second, combining this policy with the PST policy (6). In this section, we first introduce the STETT policy and discuss its advantages. Secondly, we propose the combined scheduling policy within the class of admissible scheduling policies (3).

A. *Stochastic Threshold Event-triggered Transmission*

The data triggering mechanism of the STETT policy, proposed in [18], is as follows

$$\delta_k^{st} = \begin{cases} 1, & \text{if } \frac{1}{2}e_{k|k-1}^\mathsf{T}\Theta_{k|k-1}^{-1}e_{k|k-1} > r_k \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

in which $r_k \sim \exp(\lambda_k)$ for $\lambda_k \in \mathbb{R}_{\ge 0}$ is an exponentially distributed random threshold, $e_{k|k-1} = x_k - \hat{x}_{k|k-1}$ is the predicted state estimation error and $\Theta_{k|k-1} = \mathbb{E}[e_{k|k-1}e_{k|k-1}^\mathsf{T}|\mathcal{I}_k]$ is the state error covariance.

Unlike deterministic threshold event-triggered control policies, STETT policy preserves the Gaussian property of the propagated state error at all time-steps [5]. Moreover, the transmission probability can be regulated to any desired value by tuning the random threshold parameter λ_k, which is not trivial for deterministic threshold-based policies.

However, unlike the PST policy, the STETT policy needs to implement the same state estimator as the one implemented by the controller at every time-step k. Furthermore, as we shall see shortly, although after a successful transmission $e_{k|k-1}$ is Gaussian distributed until the first new attempt to transmit, in case of collision in this transmission attempt, the distribution of the state error will become the sum of two Gaussians at the following time-step.

To see this point, note that when $\sigma_{k-1} = 1$, $e_{k-1|k-1} = 0$ and therefore $e_{k|k-1} = w_{k-1}$, which is clearly Gaussian. Assuming that the distribution of the predicted state error is Gaussian at time-step k as it is the case if $\sigma_{k-1} = 1$, but with an arbitrary covariance $\Theta_{k|k-1}$, i.e.

$$f(e_{k|k-1}|\mathcal{I}_k) = \mathcal{N}(0, \Theta_{k|k-1}), \tag{11}$$

the next lemma shows that the pdf of the predicted state error at $k+1$, in case of no data triggering ($\delta_k^{st} = 0$) is still Gaussian and in case of a data triggering and collision ($\delta_k^{st} = 1 \wedge \rho_k = 0$) is a sum of two Gaussians.

Lemma 1: Assume the distribution of the predicted state error follows (11) at time-step k, then

$$P_k = \Pr(\delta_k^{st} = 1|\mathcal{I}_k) = 1 - (1+\lambda_k)^{-\frac{n}{2}} \tag{12}$$

is the probability of the data transmission by the STETT scheduler (10) at time-step k in which n is the dimension of the state vector [18]. Moreover,

$$f(e_{k+1|k}|\delta_k^{st} = 0, \mathcal{I}_k) = \mathcal{N}(0, \hat{\Theta}_{k|k-1}) \tag{13}$$

and

$$\begin{aligned} &f(e_{k+1|k}|\delta_k^{st} = 1, \rho_k = 0, \mathcal{I}_k) = \\ &\qquad \frac{1}{P_k}\mathcal{N}(0, \Theta_{k+1|k}) - \frac{1-P_k}{P_k}\mathcal{N}(0, \hat{\Theta}_{k+1|k}) \end{aligned} \tag{14}$$

are the pdfs of the predicted state error at time-step $k+1$ in case of no data triggering ($\delta_k^{st} = 0$) and data collision ($\delta_k^{st} = 1 \wedge \rho_k = 0$), respectively, where

$$\begin{aligned} \Theta_{k+1|k} &= A\Theta_{k|k-1}A^\mathsf{T} + W, \\ \hat{\Theta}_{k+1|k} &= \frac{1}{1+\lambda_k}A\Theta_{k|k-1}A^\mathsf{T} + W. \end{aligned}$$

□

The proof of Lemma 1 is omitted due to space limitation.

Based on Lemma 1, the distribution of the state error remains Gaussian over the time period between the last successful transmission and the first collision after that. Moreover, within this time interval, in order to be in the class of the admissible triggering policies (3), the triggering probability of this policy can be set to a constant value by regulating the threshold parameter λ_k using (12). Therefore,

$$\lambda_k = 1 - (1-p)^{-\frac{2}{n}} \tag{15}$$

is an appropriate threshold parameter for having a constant transmission probability p at every time-step before the first data collision. However, when a collision happens, the distribution of the state error becomes the sum of two Gaussians. More specifically, it can be shown that in between every two successive successful transmissions, every collision doubles the number of Gaussian terms of the state error pdf [21]. On the other hand, the triggering probability when the number of Gaussian terms is more than one depends on the state error covariance and therefore, it is not trivial to have a constant triggering probability with a constant threshold parameter after the first collision instance. This motivates the proposed scheduling policy discussed in the next section.

B. Proposed Combined Event-triggered Control Policy

In this section, we propose a combined event-triggered scheduling policy $\pi = (\mu_0, \mu_1, \mu_2, \dots)$, where $\mu_k : \mathcal{I}_k \to \{0, 1\}$ and $\delta_k^{\mu} = \mu_k(\mathcal{I}_k)$, which takes the advantage of the STETT policy and at the same time guarantees a constant probability of attempting to transmit which is required by the proposed contention-based communication network to keep different control loops decoupled. Based on this policy, after every successful transmission, the scheduler triggers based on the STETT policy with a constant probability p up to the time-step at which the first collision happens. After that, the scheduler keeps triggering based on the PST policy with the same probability p until the next successful transmission time. This process is repeated in between every two successive successful transmissions.

Definition 4: Let $\bar{\ell}_k := \max\{\ell \leq k | \sigma_\ell = 1\}$ be the time of the last successful transmission before the current time-step k. Then, we can specify the combined event-triggered scheduling policy δ_k^{μ} as follows

$$\delta_k^{\mu} = \begin{cases} \delta_k^{st}, & \text{if } k = \bar{\ell}_k + 1 \text{ or if } (\delta_{\bar{\ell}_k+1}^{st}, \dots, \delta_{k-1}^{st}) = (0, \dots, 0) \\ \delta_k^{ps}, & \text{otherwise.} \end{cases} \tag{16}$$

where δ_k^{st} follows (10) with λ_k determined by (15) for a given p as the triggering probability and $\delta_k^{ps} \sim \mathrm{B}(p)$ for all time-steps. □

Remark 2: Considering Assumption 1, it can be shown that the optimal control policy for any linear dynamics (1) and the combined event-triggered scheduling policy (16) is the same as the one obtained for the PST policy in (7), (8) by following the same steps as in [21].

Remark 3: The combination of the proposed combined event-triggered scheduling policy (16) and its corresponding optimal control law (7), (8) is denoted by the combined event-triggered control (CETC) policy.

V. Main result

In this section, we state the main result of the paper. We start by establishing the first consistency property, i.e. the CETC policy outperforms the PST policy. Then in Remark 4, the second consistency property of the CETC policy is discussed. Moreover, Remark 5 addresses the mean square stability condition of the CETC policy.

Theorem 2: Suppose that Assumptions 1 and 2 hold for all the control loops using the shared contention-based communication network. Then, the average quadratic performance (2) of any control loop of this network when its scheduler-controller is operating based on the CETC policy is strictly better than the optimal control performance when the scheduler is operating based on the PST policy (base policy), i.e.,

$$J^{\pi} < J^{ps}.$$

□

The proof of Theorem 2 is omitted due to space limitation.

Remark 4: If the process disturbances are zero after a successful transmission at time-step k, i.e., $w_t = 0$ for $t > k$, the CETC policy does not attempt to transmit as opposed to the PST policy. This is the second desired consistency property of the ETCs which holds for the proposed CETC.

Remark 5: Based on Theorem 2, Assumption 2 can also guarantee the boundedness of the average quadratic performance when the control loop is operating based on the CETC policy and therefore, it is mean square stable.

VI. Numerical simulations

In this section, we illustrate via a numerical example that the proposed CETC policy performs indeed better in comparison to the PST policy and we assert the performance gains. Consider a scalar LTI system with $A = 0.9$, $B = 1$ and $W = 1$. Moreover, let $Q = 1$ and $R = 0.1$ be the parameters of the average quadratic performance. Then, the state feedback controller gain of this system is $K = -0.8233$. In Fig. 3, we compare the average quadratic performance of both policies for two different values of the probability that the network is free $q \in \{0.35, 0.7\}$. These plots illustrate what we have observed in Monte-Carlo simulations. For each pair of (p, q), we consider $n_{MC} = 10$ as the number of Monte-Carlo runs where for each of them, $T = 100000$ is the total number of simulation time-steps. The initial state for all simulations is assumed to be zero, i.e. $x_0 = 0$. Fig. 4 shows the performance gains of the CETC policy with respect to the purely stochastic policy, i.e. $\Delta J = J^{ps} - J^{\pi}$. As it can be seen, when the availability probability of the network (q) is higher, the performance gain obtained based on the event-triggered scheduling is also higher.

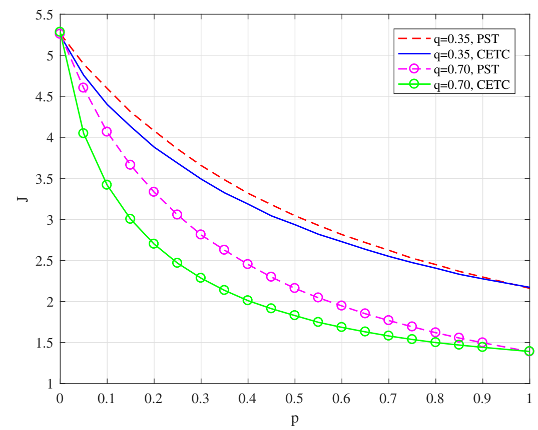

Fig. 3: Average quadratic performance of the PST and the CETC policies

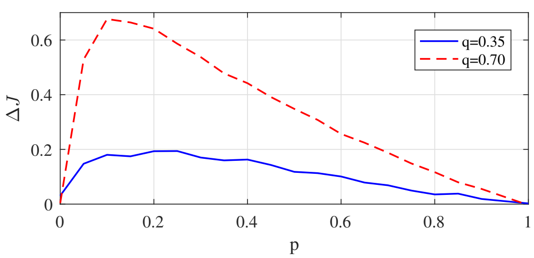

Fig. 4: Average quadratic performance gain of the CETC policy in comparison with the PST policy

VII. Conclusions

In this work, we consider a setting where the schedulers of several independent control loops transmit over a shared communication network. In order to have a scalable and easily reconfigurable design structure, we consider that the network operates based on a contention-based medium access protocol. Moreover, in order to avoid dynamic coupling between different control loops which might be generated due to state-based data collision in these kind of networks, we introduce a class of scheduling policies by which all users of this network transmit based on a random policy with a fixed transmission probability at every time-step. Then, inspired by the idea of stochastic threshold event-triggered control, we propose a novel event-based scheduling policy within the class of admissible scheduling policies. The main contribution of the present work is to establish that the performance of the proposed event-based scheduling policy is better than that of the purely stochastic transmission policy and in this sense is consistent.

References

[1] W. P. M. H. Heemels, M. C. F. Donkers, and A. R. Teel, "Periodic event-triggered control for linear systems," *Automatic Control, IEEE Transactions on*, vol. 58, no. 4, pp. 847–861, April 2013.

[2] 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.

[3] K. Gatsis, A. Ribeiro, and G. J. Pappas, "State-based communication design for wireless control systems," in *Decision and Control (CDC), 2016 IEEE 55th Conference on*. IEEE, 2016, pp. 129–134.

[4] M. H. I. Balaghi and D. J. Antunes, "Analysis of linear quadratic control loops with decentralized event-triggered sensing: Rate and performance guarantees," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, Dec 2017, pp. 6702–6707.

[5] D. Han, J. Wu, Y. Mo, and L. Xie, "On stochastic sensor network scheduling for multiple processes," *IEEE Transactions on Automatic Control*, 2017.

[6] M. H. Mamduhi, A. Molin, D. Tolić, and S. Hirche, "Error-dependent data scheduling in resource-aware multi-loop networked control systems," *Automatica*, vol. 81, pp. 209–216, 2017.

[7] M. H. Mamduhi and S. Hirche, "Try-once-discard scheduling for stochastic networked control systems," *International Journal of Control*, pp. 1–15, 2018.

[8] W. H. Heemels, M. Donkers, and A. R. Teel, "Periodic event-triggered control for linear systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 4, pp. 847–861, 2013.

[9] J. F. Kurose, *Computer networking: A top-down approach featuring the internet, 3/E*. Pearson Education India, 2005.

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

[11] 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.

[12] 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.

[13] W. Zhang, "Stabilization of networked control systems over a sharing link using aloha," in *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, vol. 1. IEEE, 2003, pp. 204–209.

[14] O. Vermesan and P. Friess, *Internet of things-converging technologies for smart environments and integrated ecosystems*. Aalborg, Denmark: River publishers, 2013.

[15] J. Stoustrup, "Plug & play control: Control technology towards new challenges," *European Journal of Control*, vol. 15, no. 3-4, pp. 311–330, 2009.

[16] R. Blind and F. Allgöwer, "Analysis of networked event-based control with a shared communication medium: Part ii–slotted aloha," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 8830–8835, 2011.

[17] ——, "On time-triggered and event-based control of integrator systems over a shared communication system," *Mathematics of Control, Signals, and Systems*, vol. 25, no. 4, pp. 517–557, 2013.

[18] F. D. Brunner, D. Antunes, and F. Allgöwer, "Stochastic thresholds in event-triggered control: A consistent policy for quadratic control," *Automatica*, vol. 89, pp. 376–381, 2018.

[19] D. J. Antunes and B. A. Khashooei, "Consistent event-triggered methods for linear quadratic control," in *Decision and Control (CDC), 2016 IEEE 55th Conference on*. IEEE, 2016, pp. 1358–1363.

[20] V. Gupta, B. Hassibi, and R. M. Murray, "Optimal lqg control across packet-dropping links," *Systems & Control Letters*, vol. 56, no. 6, pp. 439–446, 2007.

[21] M. H. Balaghi I., D. J. Antunes, M. H. Mamduhi, and S. Hirche, "An optimal lqg controller for stochastic event-triggered scheduling over a lossy communication network," *7th IFAC Workshop on Distributed Estimation and Control in Networked Systems*, 2018.