Suboptimal Event-triggered Control over Unreliable Communication Links with Experimental Validation

B. Asadi Khashooei, B. van Eekelen, D. J. Antunes and W. P. M. H. Heemels**

Abstract— **We propose an event-triggered control policy for a discrete-time linear system with unreliable actuators' and sensors' links, captured by Bernoulli packet dropout models. The proposed policy is a threshold-based policy by which transmissions occur if a weighted norm of an error state vector exceeds a threshold. The threshold and the weights of the norm depend on the underlying characteristics of the packet dropout model. Such a policy is shown to guarantee a given performance, defined in terms of a quadratic cost, while reducing transmissions. The proposed event-triggered control policy is experimentally validated in the context of remotely steering an omni-directional ground robot along a predefined trajectory over an unreliable wireless network, while keeping transmissions to a minimum.**

I. INTRODUCTION

The increasing availability of cost-effective and reliable sensors and actuators with wireless/wired networking capabilities, opens new possibilities for the manipulation of the physical world through so-called cyber devices resulting in new cyber-physical systems. In particular, there has been a growing interest in networked control systems (NCSs), for which sensors, controllers and actuators communicate via (shared) networks to obtain desired behavior for a given process.

One of the main challenges in NCS is the design of tools for efficient resource management in terms of energy, communication, and computation resources. Periodic time-triggered execution of sampling, communication and control tasks has been the main paradigm for decades. However, periodic strategies, although simple, lack the flexibility one needs to efficiently manage these resources. A recent alternative is event-triggered control (ETC, see, e.g., [1]–[15]). The main idea behind ETC is to include state or output information for the decision making of the data-transmission times in a control loop. Therefore, in a networked control setting, one can consider the ETC unit as an agent trying to create a balance between the performance of the system and the use of feedback.

The main design approaches for ETC can be classified based on the considered system's model. Some consider non-linear systems and follow emulation-based ETC approaches (later also using hybrid system approaches, see, e.g., [4], [5], [8], [16], [17]) and some focus on linear systems with stochastic disturbances [6], [7], [9], [12], [13]). The latter works often study co-design of triggering and control policies by considering a cost function, typically a quadratic function, penalizing both communication resources and control actions. We call the latter, optimization-based approaches. Optimal ETC design addresses the problem of characterizing the control and scheduling policies that minimize the considered performance criteria. However, computing optimal ETC policies is typically intractable. For example, many approaches cast the problem in a dynamic programming formulation that is limited by the curse of dimensionality. Suboptimal ETC design, on the other hand, focuses on finding simple and easy to implement control and scheduling policies that have guaranteed performance and therefore try to overcome the bottlenecks that arise with the optimal ETC design.

In the design of ETC, it is important to incorporate network artifacts such as packet dropouts, packet delays, and quantization errors, as these are present in almost all practical applications. This asks for the adaptation of the body of work in ETC to take into account these network induced artifacts. The work [18] considers the optimal control policy design for scalar systems with limited control actions under unreliable actuator links. It is shown that the optimal policy is an ETC policy in which the new control input is transmitted to the actuator if the norm of the state estimate is greater than a threshold. This threshold is obtained through a look-up table that depends on the knowledge of whether or not a dropout has happened and is determined by enumerating all possible scenarios. In [19], the triggering policy consists of a deterministic threshold-based policy and a probabilistic network access protocol determining which agent gets access to the shared network. There it was shown that if the triggering policy is aware of whether or not a packet has been dropped then the proposed policy is robust in the sense that the quadratic measure of the aggregate error remains bounded in expectation. In [20], the control of scalar Brownian motion under delta sampling, a special class of level-triggered sampling, is investigated. Furthermore, it is shown that the distortion of the deterministic sampler is always higher than the distortion due to the delta sampling. In [21], a combination of an event-based predictive control and a network compensator is shown to provide closed-loop stability of nonlinear continuous-time systems in the presence of packet dropouts. In [22], by introducing more realistic models for the communication channels with packet dropouts

*This work is supported by the Innovational Research Incentives Scheme under the VICI grant "Wireless control systems: A new frontier in automation" (no. 11382) awarded by NWO (Netherlands Organization for Scientific Research) Applied and Engineering Sciences (former Dutch Science Foundation, STW).

**The authors are with the Control System Technology group, Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands Emails: {B.Asadi.Khashooei, D.Antunes, M.Heemels}@tue.nl and b.v.eekelen@student.tue.nl

and time-delay, it has been shown that certainty equivalence is still optimal if an instantaneous error-free acknowledgment channel exists. In [23], utilizing a performance index in terms of the second moment of a scalar stochastic linear system, an ETC mechanism is designed such that, in the presence of packet dropouts, the second moment of the state converges exponentially to a desired set in finite time. It is further shown that the proposed policy in [23] under mild conditions provides guaranteed bounds over the transmission rate. In [24] the design of an output-based dynamic event-triggered mechanism for nonlinear systems subject to packet losses both with and without acknowledgment is studied. It was shown that the proposed mechanism can admit a maximum allowable number of successive packet dropouts, while still maintaining the desired stability and performance properties.

In this paper, we focus on the development of an ETC policy that provides guaranteed closed-loop performance in the presence of packet losses. To this effect, we assume that the network links connecting the sensors to the controller and the controller to actuators are subject to failures (see Figure 1). In this context, we provide a policy that takes into account UDP-like information structures at the controller side, i.e., there is no acknowledgment of actuation signal packets available to the controller [25]. We show that, taking into account a quadratic performance index, we can design a threshold event-triggered mechanism based on a weighted norm of the state estimate such that its performance is within a predefined neighborhood of the all-time transmission policy proposed in [25]. Furthermore, the applicability of the proposed policy is validated through experimental results. The experimental set-up consists of an omni-directional robot a top camera above the (soccer) field, where the robot is moving, and a controller on a host PC. Images of the robot on the soccer field are sent from the top camera to the controller via a LAN network. These images are processed to obtain the position and orientation of the robot with respect to the world frame. Then, the remote controller steers the robot along a predefined trajectory by transmitting the computed control actions to the robot via a wireless network with a UDP-like protocol.

The structure of the remainder of the paper is as follows. In Section II we formulate the problem, in Section III we explain the proposed policy. Section IV provides simulation results and in Section V the experimental setup is introduced and the results of the experiments are provided. Section VI concludes the paper. For the sake of brevity the proofs are omitted.

II. Control Structure

A schematic of the control setup is depicted in Figure 1. The plant is a linear discrete-time dynamical system

$$x_{k+1} = Ax_k + B\hat{u}_k + v_k, \quad k \in \mathbb{S} \tag{1}$$

where $\mathbb{S} = \{0, 1, \ldots, N-1\}$, $N \in \mathbb{N}$ and $x_k \in \mathbb{R}^{n_x}$, $\hat{u}_k \in \mathbb{R}^{n_u}$ are the state and the actuation signal, respectively. Moreover, $(v_k)_{k=0}^{N-1}$ is a sequence of Gaussian i.i.d. random

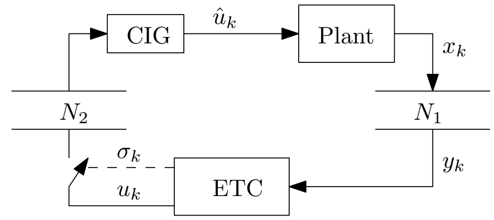

Fig. 1. Considered control structure. The sensor network, N_1, and the actuator network N_2 are subject to Bernoulli dropout failures. CIG stands for control-input generator.

vectors with zero mean and covariance Φ_v. We consider a network between the sensors and the controller N_1, and a network between the controller and the actuators N_2. In a similar setting as in [25], [26], we assume that the networks are lossy with dropout probabilities α and β, respectively. The control unit not only computes the control action u_k but also decides whether or not to transmit the computed control action based on the available information. This scheduling is modeled via a decision variable $\sigma_k \in \{0, 1\}$, $k \in \mathbb{S}$, where $\sigma_k = 1$ denotes the occurrence of a transmission and $\sigma_k = 0$ a non-transmission event. At the actuator side, we consider a control-input-generator (CIG) that sets the actuation signal to zero when no control input is received from the controller. Let $(\alpha_k)_{k\in\mathbb{S}}$ and $(\beta_k)_{k\in\mathbb{S}}$ be i.i.d. Bernoulli processes which model the unreliable nature of the link from the controller to the actuators and sensors to controller, respectively. Then,

$$\hat{u}_k = \alpha_k \sigma_k u_k, \quad k \in \mathbb{S} \tag{2}$$

and the system (1) can be modeled as

$$\begin{aligned} x_{k+1} &= Ax_k + \alpha_k \sigma_k B u_k + v_k \\ y_k &= \begin{cases} x_k, & \beta_k = 1, \\ \emptyset, & \text{otherwise.} \end{cases} \end{aligned} \tag{3}$$

The packet dropouts have the following probability distribution

$$\begin{aligned} P[\alpha_k = 1] &= 1 - \alpha = \bar{\alpha} \\ P[\alpha_k = 0] &= \alpha \end{aligned} \tag{4}$$

and

$$\begin{aligned} P[\beta_k = 1] &= 1 - \beta = \bar{\beta} \\ P[\beta_k = 0] &= \beta. \end{aligned} \tag{5}$$

We assume that $(\alpha_k)_{k\in\mathbb{S}}$ and $(\beta_k)_{k\in\mathbb{S}}$ are independent processes, also independent of the disturbance process $(v_k)_{k\in\mathbb{S}}$, and the initial state x_0. Moreover, note that $y_k \in \mathbb{R}^{n_x}$ denotes the accessible state of the plant to the controller and the symbol $y_k = \emptyset$ is used to denote the absence of information when a packet drop occurs.

To design and analyze the controller, we consider the following finite-horizon quadratic performance criterion

$$J = \mathbb{E}[x_N^T Q_N x_N + \sum_{k=0}^{N-1} x_k^T Q x_k + \hat{u}_k^T R \hat{u}_k] \tag{6}$$

where Q and R are positive definite weighting matrices of proper dimensions. To penalize transmissions, we follow a similar approach as in [13] and introduce an additive penalty term to the performance index (6). Thus we obtain

$$V = \mathbb{E}[x_N^T Q_N x_N + \sum_{k=0}^{N-1} x_k^T Q x_k + \hat{u}_k^T R \hat{u}_k + \lambda \sigma_k] \quad (7)$$

where $\lambda > 0$ characterizes the penalty on transmissions.

We define a control and scheduling policy π as a set of functions

$$\pi = \{(\mu_0^\sigma(I_0), \mu_0^u(I_0)), \ldots, (\mu_{N-1}^\sigma(I_{N-1}), \mu_{N-1}^u(I_{N-1}))\}. \quad (8)$$

In particular, the available information vector I_k (to be specified) at time $k \in \mathbb{S}$, is mapped to the scheduling variable and control action (σ_k, u_k) as

$$(\sigma_k, u_k) = \mu_k(I_k) = (\mu_k^\sigma(I_0), \mu_k^u(I_k)). \quad (9)$$

We consider that both networks are unreliable and a UDP-like protocol in which there is no acknowledgment signal is applied. The information vector can be represented as

$$\begin{aligned} I_k^{UDP} &= (I_{k-1}^{UDP}, y_k, \beta_k, u_{k-1}, \sigma_{k-1}), \quad k \in \mathbb{S} \backslash \{0\} \\ I_0^{UDP} &= (y_0, \beta_0). \end{aligned} \quad (10)$$

We denote the average triggering rate as

$$R_t = \frac{1}{N} \mathbb{E}[\sum_{k=0}^{N-1} \sigma_k]. \quad (11)$$

The optimal control and scheduling policy design focuses on finding the policy π_{opt} to minimize (7). This problem is a mixed-integer programing and therefore obtaining an optimal solution is computationally not tractable. Therefore, in this paper, we focus on finding a *good* policy π that is simple to implement and achieves a performance in terms of (6), which is within a guaranteed analytical bound. We propose a control policy corresponding to the aforementioned information structure. The proposed policy is built upon the roll-out algorithm (see, e.g., [27]) in the context of approximate dynamic programming considering a base triggering policy of all-time transmission $\sigma_k = 1,\ k \in \mathbb{S}$, and optimal control policy. The provided guaranteed analytical bounds are obtained using the performance of this base policy.

III. ETC FOR UNRELIABLE COMMUNICATION LINKS WITH UDP-LIKE PROTOCOL

We assume that both actuator and sensor network links are prone to failure, i.e., $\beta > 0,\ \alpha > 0$, and there is no acknowledgment of successful transmission in the actuator network. Then the available information at the controller is characterized by (10). In this setting, if transmissions are triggered at all time instants i.e. $\sigma_k = 1,\ k \in \mathbb{S}$, then the problem reduces to the optimal control design investigated in [25]:

$$\begin{aligned} u_k &= L_k \hat{x}_k \\ L_k &= -(R + B^T(K_{k+1} + \alpha\beta P_{k+1})B)^{-1} B^T K_{k+1} A \\ P_k &= \bar{\alpha} A^T K_{k+1} B (R + B^T(K_{k+1} + \alpha\beta P_{k+1})B)^{-1} \\ &\quad \times B^T K_{k+1} A + \beta A^T P_{k+1} A \\ K_k &= A^T K_{k+1} A - P_k + \beta A^T P_{k+1} A + Q \end{aligned} \quad (12)$$

where $K_N = Q_N,\ P_N = 0$ and $e_k := x_k - \hat{x}_k$ with $\hat{x}_k$ the state estimate whose dynamics are governed by the optimal estimator (see, [25, eq. 17]),

$$\hat{x}_k = \begin{cases} A\hat{x}_{k-1} + \bar{\alpha} B u_{k-1} & \text{if } \beta_k = 0 \\ x_k & \text{if } \beta_k = 1. \end{cases} \quad (13)$$

Lemma 1: The control policy (12), (13) for all-time transmission with a UDP-like network protocol results in the overall performance of

$$V_0^{\pi_{all}}(I_0^{UDP}) = J_0^{\pi_{all}}(I_0^{UDP}) + N\lambda \quad (14)$$

where

$$J_0^{\pi_{all}}(I_0^{UDP}) = \mathbb{E}[x_0^T K_0 x_0 | I_0^{UDP}] + \mathbb{E}[e_0^T P_0 e_0 | I_0^{UDP}] + \sum_{i=0}^{N-1} \mathbb{E}[v_i^T (K_{i+1} + \beta P_{i+1}) v_i]. \quad (15)$$

□

In the following theorem, the proposed ETC policy under the information structure (10) is introduced. The goal of the ETC design is to find a policy that provides a guaranteed performance under unreliable communication links, while reducing the number of transmissions in the actuator network.

Theorem 1: Consider the system (3) with control and scheduling policy π defined as

$$(u_k, \sigma_k) = \begin{cases} (L_k \hat{x}_k, 1), & \text{if } \hat{x}_k^T \Gamma_k \hat{x}_k > \lambda \\ (\emptyset, 0), & \text{otherwise} \end{cases} \quad (16)$$

$$\begin{aligned} \Gamma_k &= \bar{\alpha} A^T K_{k+1} B (R + B^T(K_{k+1} + \alpha\beta P_{k+1})B)^{-1} \\ &\quad \times B^T K_{k+1} A \end{aligned}$$

where

$$\hat{x}_k = \begin{cases} A\hat{x}_{k-1} + \bar{\alpha} B \sigma_{k-1} u_{k-1} & \text{if } \beta_k = 0 \\ x_k & \text{if } \beta_k = 1. \end{cases} \quad (17)$$

Then the cost of the policy π, denoted by $J_0^\pi(I_0^{UDP})$, satisfies

$$J_0^\pi(I_0^{UDP}) \le J_0^{\pi_{all}}(I_0^{UDP}) + N\lambda. \quad (18)$$

IV. SIMULATION RESULTS

We consider a discretized model of a double integrator controlled over two networks as depicted in Figure 1. The discretized system with the sampling time of 0.1 (sec) and the cost parameters are given by

$$A = \begin{bmatrix} 1 & 0.1 \\ 0 & 1 \end{bmatrix},\ B = \begin{bmatrix} 0.005 \\ 0.1 \end{bmatrix},$$

$$\Phi_v = 0.01, \quad N = 100, \quad Q_N = Q = I, \quad R = 1.$$

In this example, we consider the drop-out probabilities of $\beta = 0.6$ and $\alpha = 0.3$ for the sensor and the actuator networks, respectively. Figure 2 shows the simulation results for the proposed policy with $\lambda \in [0,\ 1]$ based on Monte Carlo simulation of 600 realizations. As can be seen the proposed policy led to a system's operation within the guaranteed analytical performance bounds.

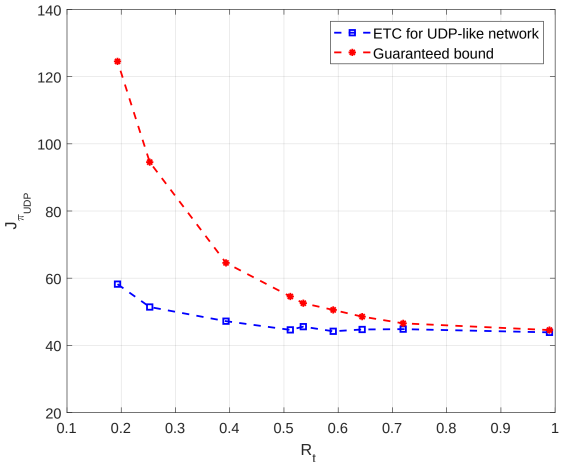

Fig. 2. Performance of the proposed ETC policy under unreliable network links. The vertical axis represents the performance in terms of (6) for the network with a UDP-like protocol. The horizontal axis represents the average triggering rate as defined in (11).

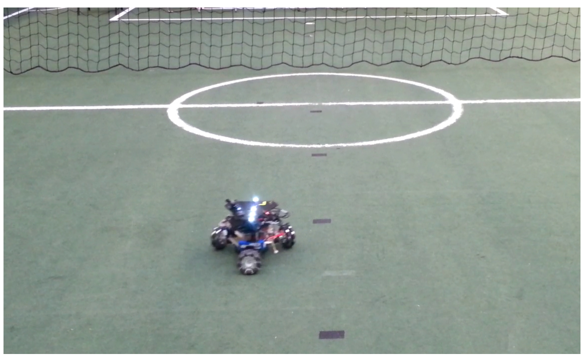

Fig. 3. The omni-directional robot used for the experiments.

V. Experimental validation for remote control of a ground robot

In this section, we use the developed algorithm for model-based tracking and remote regulation of a ground robot towards a predefined trajectory. The experimental set-up consists of an omni-directional robot, see Figure 3, a top camera above a soccer field, where the robot is moving and a controller on a host PC. Images of the robot on the soccer field are sent from the top camera to the controller, these images are processed to obtain the position and orientation of the robot with respect to the world frame. Then based on the calculated position control actions are computed as will be discussed in the sequel.

Figure 4 shows a schematic representation of the control structure. The plant represents the omni-directional robot, whose dynamics can be described by a linear model The robot model is given by the following discrete-time linear

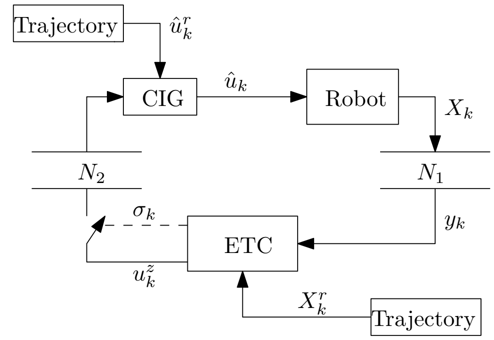

Fig. 4. Block diagram of a controller which combines state feedback and feedforward from a trajectory generator. The feedforward command u_k^r along with the desired state X_k^r are calculated a priori and stored on both the robot and the ETC unit. The feedback controller uses the received state y_k and the reference trajectory X_k^r to compute a corrective input u_k^z.

time-invariant system

$$\underbrace{\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \psi_{k+1} \end{bmatrix}}_{X_{k+1}} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_k \\ y_k \\ \psi_k \end{bmatrix}}_{X_k} + T_s \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} \hat{u}_{x,k} \\ \hat{u}_{y,k} \\ \hat{u}_{\psi,k} \end{bmatrix}}_{\hat{u}_k} + v_k$$
$$y_k = \begin{cases} X_k, & \beta_k = 1, \\ \emptyset, & \text{otherwise}, \end{cases} \tag{19}$$

where T_s is the sampling time, which is set to 0.1 second. The control input $\hat{u}_k$ consists of a feedforward term $\hat{u}_k^r$ and a feedback term $\hat{u}_k^z$. The feedforward term is computed from the desired model-based trajectory $X_k^r,\ k \in \mathbb{S}$, and obtained through

$$X_{k+1}^r = AX_k^r + B\hat{u}_k^r. \tag{20}$$

The feedback term, on the other hand, regulates the robot towards the desired trajectory in the presence of disturbances. Since the system is linear we can use the proposed ETC policy to design the feedback control input based on the error between the position of the robot X_k and the reference, X_k^r which has the dynamics

$$z_{k+1} = Az_k + B\hat{u}_k^z + v_k$$

where $z_k = X_k - X_k^r$. We assume that both the remote controller and the robot have access to the feedforward control inputs and the controller only sends the feedback input to the robot. Therefore, the control input is obtained by

$$\hat{u}_k = \hat{u}_k^z + \hat{u}_k^r \tag{21}$$

where

$$\hat{u}_k^z = \sigma_k \alpha_k u_k^z. \tag{22}$$

Therefore, if no control input is received by the robot either because of packet dropout $\alpha_k = 0$ or because the triggering condition is not active, i.e., $\sigma_k = 0$, then the feedforward term $\hat{u}_k^r$ is still applied to the actuators, i.e., only the feedback term $\hat{u}_k^z$ of the input $\hat{u}_k$ is set to zero.

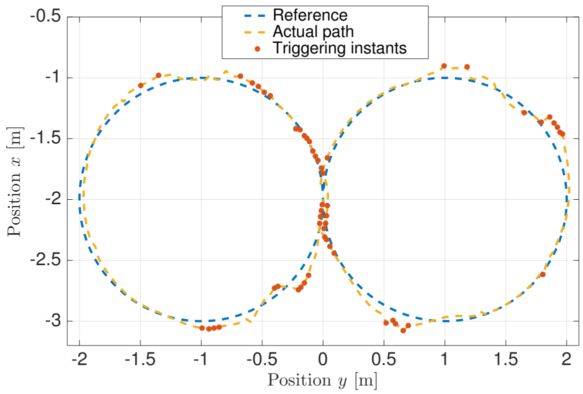

Fig. 5. The tracking in the x, y-plane for the proposed event-triggered mechanism (16) with threshold $\lambda = 0.01$, only the tracking for the first perambulation is shown. The instants at which a triggering occurs are depicted with red dots.

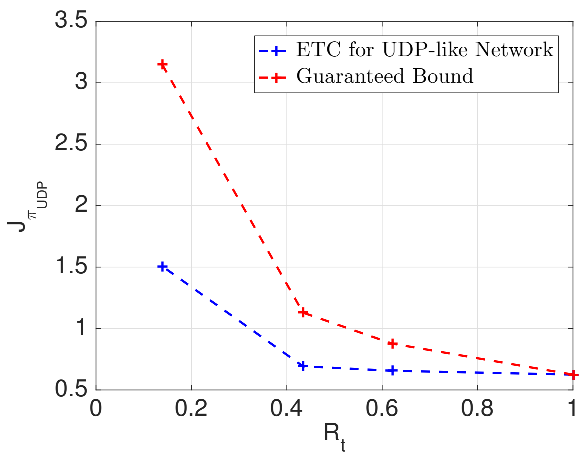

Fig. 6. Performance of the proposed ETC in the considered experimental setup where $J = \mathbb{E}[z_N^T Q_N z_N + \sum_{k=0}^{N-1} z_k^T Q z_k + \hat{u}_k^{z^T} R \hat{u}_k^z]$.

Experimental results

In the experiments, we consider the performance index (6) with

$$Q = I_3, \quad Q_N = I_3, \quad R = I_3, \quad N = 250.$$

The actual percentage of packets that was lost in the actuator network is approximately 1%. Therefore, to illustrate the power of the proposed policy, randomly induced packet drops were introduced to acquire a drop-out probability of $\alpha = 0.1$ (accounts for 10% drop-out rate). In the sensor network the packet drop-outs were mainly due to missing the computation deadline. This leads to a dropout probability corresponding to $\beta = 0.25$. For this experiment we only consider the UDP-like protocol thus no acknowledgment is available in the actuator network.

Since the computation of a (new) feedback control value can not be done instantaneously at sampling times, we introduce a unit delay for applying controller actions and, therefore, we update the state estimation $\hat{z}_k$ by prediction i.e.,

$$u_k = L_k(A\hat{z}_{k-1} + \bar{\alpha} B u_{k-1}).$$

The reference trajectory consists of two tangent circles with radius of 1 meter as depicted by blue dashes in Figure 5. Moreover, the real trajectory and transmission instants of the remote robot are also depicted in Figure 5.

Figure 6 shows the experimental results for four different values of $\lambda \in \{0,\ 0.001,\ 0.002,\ 0.01\}$. Each experiment was carried out for 5 rounds and the obtained performance is averaged over these experiments. Interestingly, the performance of the proposed ETC policy, J^{π}, is always below the guaranteed bounds, $J^{\pi_{all}}$, which validates the proposed scheme in the experimental set-up.

VI. Conclusions

A suboptimal ETC policy with guaranteed performance under unreliable actuator and sensor links was provided in this paper. The proposed policy belongs to the class of threshold-based policies, whose parameters are influenced by the underlying characteristics of the packet dropout behaviour in the communication network and the cost of communication in the considered quadratic performance index. An experimental setup for validation of the proposed algorithm has been developed. In the experiment a remote controller is used to steer an omni-directional ground robot, along a predefined trajectory. The conducted experiments validated the theoretical results and the applicability of the proposed ETC policy.

References

[1] J. K. Yook, D. M. Tilbury, and N. R. Soparkar, "Trading computation for bandwidth: Reducing communication in distributed control systems using state estimators," *IEEE transactions on control systems technology*, vol. 10, no. 4, pp. 503–518, 2002.

[2] W. P. M. H. Heemels, J. H. Sandee, and P. P. J. van den Bosch, "Analysis of event-driven controllers for linear systems," *International Journal of Control*, vol. 81, no. 4, pp. 571–590, 2008.

[3] T. Henningsson, E. Johannesson, and A. Cervin, "Sporadic event-based control of first-order linear stochastic systems," *Automatica*, vol. 44, no. 11, pp. 2890 – 2895, 2008.

[4] M. Abdelrahim, R. Postoyan, J. Daafouz, and D. Nesic, "Stabilization of nonlinear systems using event-triggered output feedback controllers," *Automatic Control, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015.

[5] V. S. Dolk, D. P. Borgers, and W. P. M. H. Heemels, "Output-based and decentralized dynamic event-triggered control with guaranteed $mathcalL_p$- gain performance and zeno-freeness," *IEEE Transactions on Automatic Control*, vol. 62, no. 1, pp. 34–49, Jan 2017.

[6] C. Ramesh, H. Sandberg, L. Bao, and K. H. Johansson, "On the dual effect in state-based scheduling of networked control systems," in *American Control Conference (ACC), 2011*, June 2011, pp. 2216–2221.

[7] B. Demirel, V. Gupta, and M. Johansson, "On the trade-off between control performance and communication cost for event-triggered control over lossy networks," in *Control Conference (ECC), 2013 European*, July 2013, pp. 1168–1174.

[8] J. Lunze and D. Lehmann, "A state-feedback approach to event-based control," *Automatica*, vol. 46, no. 1, pp. 211 – 215, 2010.

[9] A. Molin and S. Hirche, "Structural characterization of optimal event-based controllers for linear stochastic systems," in *Proceedings of the IEEE International Conference on Decision and Control (CDC 2010)*, 2010, pp. 3227–3233.

[10] D. Antunes and W. P. M. H. Heemels, "Rollout event-triggered control: Beyond periodic control performance," *Automatic Control, IEEE Transactions on*, vol. 59, no. 12, pp. 3296–3311, Dec 2014.

[11] W. Wu, S. Reimann, D. Gorges, and S. Liu, "Suboptimal event-triggered control for time-delayed linear systems," *Automatic Control, IEEE Transactions on*, vol. 60, no. 5, pp. 1386–1391, May 2015.

[12] J. Araújo, A. Teixeira, E. Henriksson, and K. H. Johansson, "A down-sampled controller to reduce network usage with guaranteed closed-loop performance," in *53rd IEEE Conference on Decision and Control*. IEEE, 2014, pp. 6849–6856.

[13] Y. Xu and J. P. Hespanha, "Optimal communication logics in networked control systems," in *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, vol. 4, Dec 2004, pp. 3527–3532 Vol.4.

[14] E. Garcia, P. J. Antsaklis, and L. A. Montestruque, *Model-Based Control of Networked Systems*, ser. Systems & Control: Foundations & Applications. Springer International Publishing, 2014.

[15] M. H. Mamduhi, D. Tolic, A. Molin, and S. Hirche, "Event-triggered scheduling for stochastic multi-loop networked control systems with packet dropouts," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, Dec 2014, pp. 2776–2782.

[16] M. C. F. Donkers and W. P. M. H. Heemels, "Output-based event-triggered control with guaranteed $\mathcal{L}_\infty$-gain and improved and decentralized event-triggering," *Automatic Control, IEEE Transactions on*, vol. 57, no. 6, pp. 1362–1376, June 2012.

[17] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *Automatic Control, IEEE Transactions on*, vol. 52, no. 9, pp. 1680–1685, Sept 2007.

[18] P. Bommannavar and T. Basar, "Optimal control with limited control actions and lossy transmissions," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*. IEEE, 2008, pp. 2032–2037.

[19] M. H. Mamduhi, D. Tolić, A. Molin, and S. Hirche, "Event-triggered scheduling for stochastic multi-loop networked control systems with packet dropouts," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*. IEEE, 2014, pp. 2776–2782.

[20] M. Rabi and K. H. Johansson, "Scheduling packets for event-triggered control," in *Control Conference (ECC), 2009 European*. IEEE, 2009, pp. 3779–3784.

[21] P. Varutti and R. Findeisen, "Event-based nmpc for networked control systems over udp-like communication channels," in *American Control Conference (ACC), 2011*. IEEE, 2011, pp. 3166–3171.

[22] A. Molin and S. Hirche, "Suboptimal event-based control of linear systems over lossy channels," *IFAC Proceedings Volumes*, vol. 43, no. 19, pp. 55–60, 2010.

[23] P. Tallapragada, M. Franceschetti, and J. Corts, "Event-triggered stabilization of scalar linear systems under packet drops," in *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sept 2016, pp. 1173–1180.

[24] V. S. Dolk and W. P. M. H. Heemels, "Dynamic event-triggered control under packet losses," *Automatica*, to appear.

[25] O. C. Imer, S. Yksel, and T. Basar, "Optimal control of LTI systems over unreliable communication links," *Automatica*, vol. 42, no. 9, pp. 1429 – 1439, 2006.

[26] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry, "Foundations of control and estimation over lossy networks," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 163–187, Jan 2007.

[27] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Athena Scientific, 2005.