

# Resilient Event Triggered Systems with Limited Communication

Lichun Li, Bin Hu and Michael Lemmon

**Abstract**—A resilient control system is one that maintains state awareness and an accepted level of operational normalcy in response to unexpected disturbances. There has recently been great interest in event-triggered control for networked systems. It is unclear, however, whether event-triggering is appropriate for resilient control because of the sporadic nature of the feedback. This paper examines the bit-rates needed to realize event-triggered controls that are resilient to transient faults. Using techniques from dynamically quantized control, we derive sufficient resilient bit-rates nonlinear scalar systems with affine controls and disturbances. For linear systems, these sufficient rates can be necessary for resilience. The results in this paper suggest that, at least for transient faults, resilient control is indeed achievable using event-triggered feedback.

## I. INTRODUCTION

Resilient operation of critical systems is of great societal importance. Failures of critical civil infrastructure, such as the power grid or transportation network, will result in economic losses and societal disruptions. The scale of these disruptions will be grow larger and larger as more systems are integrated over public networks. These public systems also become more vulnerable to malicious attacks as system size increases. It is therefore crucial that the controls strategies used for these systems be resilient to these threats.

A resilient control system is one that maintains state awareness of threats and anomalies [1]. In particular, we expect such control systems to provide guarantees on a system's return to operational normalcy in the presence of disturbances generated by such threats and anomalies. The impact of such threats can be modeled as a *stop faults* which change the structure of the system [2]. Such threats may also be modeled as *transient faults* that result in a discontinuous step change in the plant's state. This paper examines resilience to such transient faults. We consider a nonlinear scalar system whose operational range can be partitioned into a *safe* and *unsafe* region. The safe operational region is a neighborhood of the origin (equilibrium point) over which the controller ensures a pre-specified performance objective. Disturbances due to threats are impulses of *unknown magnitude* that may drive a system into its unsafe region. Within the unsafe region it is no longer possible to guarantee system performance levels. One may, however, be able to use an *emergency* control action that guarantees the system's return to the safe region in finite time. The main result of this paper characterizes lower bounds on those bit rates ensuring resilient operation of *quantized event-triggered* control systems.

The authors are with Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556, USA. [ll13](mailto:ll13@nd.edu), [bhu2](mailto:bhu2@nd.edu), [lemmon@nd.edu](mailto:lemmon@nd.edu)

Event triggered systems have shown their potential to conserve communication resources while preserving system performance. In event triggered systems, transmission occurs when the output of the system exceeds a threshold. Recent experiments have shown that event triggered systems use fewer communication resources than time triggered systems having comparable performance [3], [4], [5], [6]. The key reason is that event triggering makes use of online information to decide transmission time, while time triggering uses a fixed transmission interval based on the worst case scenario.

Many embedded system engineers, however, favor the use of time-triggered over event-triggered control architectures. A major objection to the use of event-triggered control is that it tends to generate *sporadic* information streams. In these sporadic streams, the time between consecutive transmissions changes in a time-varying manner. This lack of predictability is seen as an obstacle to resilient operation. This concern may appear valid on the surface, but in fact event-triggered systems do provide some degree of predictability in that the minimum time between successive transmissions is usually bounded away from zero. In this regard, event-triggering can be seen as providing a minimum level of feedback connectivity, with additional information being transmitted over the channel when the system is driven away from its equilibrium point. A key question is how small this feedback information rate can be made while still assuring resilient operation?

The main results in this paper characterize an upper bound on the minimum bit-rates required for resilient behavior. These results are obtained for scalar nonlinear systems whose control and disturbances enter in an affine manner. The control input,  $u$ , is an impulse train whose impulses are applied when sensor data is transmitted over a communication channel to the system's actuator. Disturbance inputs are also impulses that force a jump in the plant's state as the result of some transient fault. We assume that the magnitude of these faults is *unknown*. The occurrence of such fault forces the system out of its safe operating region and into an unsafe region. The main result of this paper establishes upper bounds on the information bit rate between sensor and actuator required to 1) eventually force an unsafe system back into its safe operating region and 2) maintain a specified performance level within the safe performance region.

## II. MATHEMATICAL PRELIMINARIES

Let  $\mathbb{R}^n$  denote the linear space of real  $n$ -vectors. Let  $\mathbb{R}^+$  denote the set of non-negative reals. The infinity norm of a vector  $x \in \mathbb{R}^n$  will be denoted as  $|x|$ . Given the real-valued

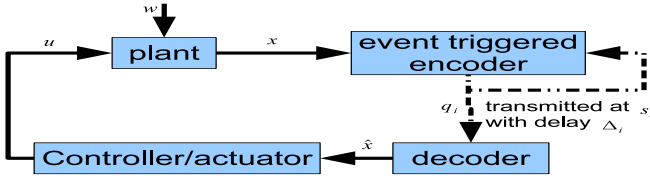


Fig. 1. System structure

function  $x(\cdot) : \mathbb{R}^+ \rightarrow \mathbb{R}^n$ , we let  $x(t)$  denote the value  $x$  takes at time  $t \in \mathbb{R}^+$ .

Let  $\Omega$  be a closed and bounded subset of  $\mathbb{R}^n$ . We say  $f(\cdot) : \Omega \rightarrow \mathbb{R}^n$  is Lipschitz on  $\Omega$  if for any  $x, y \in \Omega$ , we know there exists a constant  $L \geq 0$  such that

$$|f(x) - f(y)| \leq L|x - y|,$$

where  $L$  is called the Lipschitz constant.

A function  $\alpha(\cdot) : \mathbb{R}^+ \rightarrow \mathbb{R}^+$  is class  $\mathcal{K}$  if it is continuous, strictly increasing and  $\alpha(0) = 0$ . A function  $\beta : \mathbb{R}^+ \times \mathbb{R}^+ \rightarrow \mathbb{R}^+$  is class  $\mathcal{KL}$  if  $\beta(\cdot, t)$  is class  $\mathcal{K}$  for each fixed  $t \geq 0$  and  $\beta(r, t)$  decreases to 0 as  $t \rightarrow \infty$  for each fixed  $r \geq 0$ .

Let  $\Omega_a \subseteq \mathbb{R}^n$  be a compact set containing the origin. Suppose  $x(t) \in \mathbb{R}^n$  is the state trajectory satisfying

$$\dot{x} = f(x).$$

Let  $t_0$  be the time when  $x(t)$  first enters  $\Omega_a$ . We say that  $\Omega_a$  is an *absorbing region* of  $f$  with performance level  $\beta(t - t_0)$ , if for all  $t \geq t_0$ ,

$$|x(t)| \leq \beta(t - t_0),$$

where  $\beta(\cdot)$  is a strictly decreasing function. The magnitude of a region  $\Omega$ , denoted by  $|\Omega|$ , is defined as

$$|\Omega| = \max\{|x|, \text{for all } x \in \Omega\}$$

### III. SYSTEM SETUP

The system structure is shown as in Figure 1. The plant is a nonlinear scalar system whose state trajectory  $x : \mathbb{R}^+ \rightarrow \mathbb{R}$  satisfies the following initial value problem,

$$\dot{x}(t) = f(x(t)) + u(t) + w(t) \quad (1)$$

for all  $t \geq 0$  with  $x(0) = x_0$ . We assume that  $f(\cdot) : \mathbb{R} \rightarrow \mathbb{R}$  is a locally Lipschitz about the origin with Lipschitz constant  $L$  with  $f(0) = 0$ . The input signal  $u$  is a train of control impulses and the input signal  $w$  is a train of exogenous disturbance impulses generated by transient system faults.

The disturbance signal,  $w$ , is an impulse train of the form,

$$w(t) = \sum_{i=1}^{\infty} \omega_i \delta(t - h_i)$$

where  $h_i$  denotes the  $i$ th consecutive *fault* time and  $\omega_i \in \mathbb{R}$  represents the *magnitude* of the  $i$ th fault. The main feature of this fault model is that  $\omega_i$  is not known or bounded in an *a priori* manner.

The control input,  $u$ , is generated by the controller shown in figure 1. In this system, a subsystem called the *encoder* samples the plant's state at discrete time instants  $s_i$  for

$i = 1, 2, \dots, \infty$ . The  $i$ th consecutive sampled state,  $x(s_i)$ , is encoded as a codeword  $q_i \in \{1, 2, \dots, Q\}$ . The event-triggered encoder then transmits this codeword over the channel at time  $s_i$  with a delay  $\Delta_i \geq 0$ . The transmitted codeword is received at time instant  $r_i = s_i + \Delta_i$ . Upon receiving this codeword, the decoder generates the control update,  $\hat{x}_i \in \mathbb{R}$ . The controller/actuator then takes this update and applies it to the plant through an impulse at time  $r_i$ . The final control signal input therefore takes the form of an impulse train,

$$u(t) = - \sum_{i=1}^{\infty} \hat{x}_i \delta(t - r_i)$$

where  $\hat{x}_i$  is the  $i$ th consecutive control command applied at time instant  $r_i$ .

It will be convenient to restrict the system function,  $f$ , so that

$$xf(x) > 0 \text{ for all } x \in \mathbb{R}, \quad (2)$$

$$f(|x|) \text{ and } -f(-|x|) \text{ are class } \mathcal{K} \quad (3)$$

The first assumption follows from our Lipschitz assumption and the second assumption can be relaxed at the cost of degrading the bit-rate bounds derived below. Finally, we assume there exists a real number  $x_a$  such that for all  $|x| > x_a$  we can uniformly bound  $|f(x)|$ . In other words, we assume there exist positive constants  $L$  and  $\bar{L}$  such that

$$|f(x)| \leq \begin{cases} L|x| & \text{if } |x| \leq x_a \\ \bar{L} & \text{if } |x| > x_a \end{cases} \quad (4)$$

We refer to the set  $\Omega_s \equiv \{x \in \mathbb{R} : |x| \leq x_a\}$  as the *safe region* and  $\Omega_u \equiv \{x \in \mathbb{R} : |x| > x_a\}$  as the *unsafe region*. The motivation for this terminology is as follows. For the safe region, the Lipschitz constant,  $L$ , provides an accurate upper bound on the system's rate of growth. In the unsafe region, however, this Lipschitz approximation is too conservative. The unsafe region represents that region in which the plant's open-loop rate of change begins to saturate due to physical limitations within the system.

The safe regime is a regime under which a desired level of performance is guaranteed (as shown in the right part of figure 2). In this paper, we specify this desired performance as exponential stability. Let  $t_0$  be the initial time when the system switches to safe regime, and  $t_f$  be the terminal time when the system exits safe regime. For all  $t \in [t_0, t_f]$ , the system state  $x(t)$  satisfies

$$|x(t)| \leq \rho x_a e^{-\alpha(t-t_0)}, \text{ for some } \alpha \geq 0 \text{ and } \rho \geq 1 \quad (5)$$

When the performance in (5) is violated, the system switches to unsafe regime. Unsafe regime is a regime under which the desired level of performance is not guaranteed. Let  $\tau_0$  and  $\tau_f$  be the initial time and terminal time of the unsafe regime. There must exist some time  $t \in [\tau_0, \tau_f]$  such that

$$|x(t)| > \rho x_a e^{-\alpha(t-\tau_0)}.$$

To be resilient, we require that the system is able to come back to the safe regime from unsafe regime in finite time.

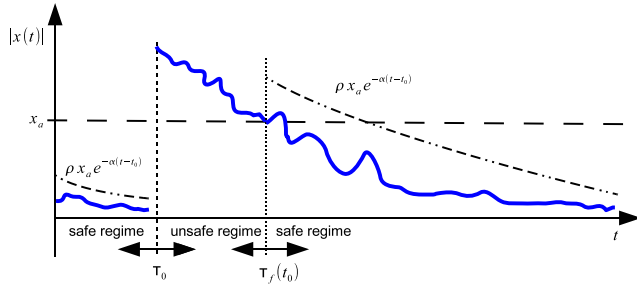


Fig. 2. Safe regime, unsafe regime and resilience

To be more specific, we define the resilience of a system as below.

**Definition 3.1:** A system is resilient if for all state  $x(\tau_0)$  satisfying  $|x(\tau_0)| \in (x_a, \infty)$ , there is a finite time  $\tau_f \in (0, \infty)$  such that

$$|x(\tau_f)| \leq x_a. \quad (6)$$

We use the term *resilience* to refer to a system's ability to return to the safe region in *finite* time. What makes resilience distinct from *robustness* is that the transient faults have an arbitrarily large magnitude, rather than just a bounded magnitude. Because the magnitude of the fault can be arbitrarily large, the controller/actuator doesn't know exactly what control is needed to return the state to the origin. So we propose having the controller apply an *emergency* or *safe* control step. The main result of this paper characterizes the rate at which this emergency control must be applied to assure that the system is resilient; i.e. that it returns to its safe region in finite time.

#### IV. NECESSARY BIT-RATE AND SUFFICIENT BIT-RATE IN SAFE REGIME

##### A. Necessary Bit-Rate in Safe Regime

Let's assume that at time  $t_0$ , the system enters the safe regime. Define  $\phi(x_0, t - t_0)$  as the zero-input behavior of plant (1), where  $x_0$  and  $t_0$  are the initial state and initial time. With the assumptions in (2) and (3), the zero-input behavior of plant (1) has some nice properties which are given in the following lemma.

**Lemma 4.1:** If equation (2) and (3) hold, then for some small enough time interval  $\varepsilon > 0$ , we have

$$\phi(-x, \varepsilon) = -\phi(x, \varepsilon), \quad (7)$$

$$\phi(x_1, \varepsilon) \geq \phi(x_2, \varepsilon), \text{ if } x_1 \geq x_2 \geq 0. \quad (8)$$

Now, let's give the necessary bit-rate to maintain exponential stability in safe regime in the next theorem. The basic idea is similar to the idea used in dynamic quantization area [7], [8].

**Theorem 4.2:** Let  $t_0$  be the initial time when the system switch to safe regime, and  $\beta(t) = \rho x_a e^{-\alpha(t-t_0)}$ . If the performance level in equation (5) is guaranteed for all  $t \geq t_0$ , then the instantaneous bit-rate  $r(t)$  at time  $t$  satisfies

$$r(t) \geq r_s(t) = \max \left\{ \frac{1}{\ln 2} \left( \frac{f(\beta(t))}{\beta(t)} + \alpha \right), 0 \right\}. \quad (9)$$

$r_s$  is called the necessary stabilizing bit-rate in safe regime.

**Proof:** Since performance level (5) is guaranteed for any  $t \geq t_0$ , we know that for any  $t$ ,  $x(t)$  is within interval  $[-\beta(t), \beta(t)]$ . From Lemma 4.1, we know that after a small enough time interval  $\varepsilon$ ,  $x(t + \varepsilon)$  should lie in another interval  $\mathcal{R} = [-\phi(\beta(t), \varepsilon) + \mu, \phi(\beta(t), \varepsilon) + \mu]$ , where  $\mu$  is the total amount of control input applied during interval  $[t, t + \varepsilon]$ . So the volume of  $\mathcal{R}$  is

$$V(\mathcal{R}) = 2\phi(\beta(t), \varepsilon). \quad (10)$$

We divide the interval  $\mathcal{R}$  into  $N$  parts, and let  $\mathcal{L}_j$  indicate the  $j$ th part. so the total volume of these parts satisfies

$$\sum_{j=1}^N V(\mathcal{L}_j) = V(\mathcal{R}). \quad (11)$$

Since the performance level (5) is guaranteed for all  $t \geq t_0$ , we know that

$$V(\mathcal{L}_j) \leq 2\beta(t + \varepsilon). \quad (12)$$

Therefore, from equation (11), (12) and (13), we arrive at the inequality that

$$N \geq \frac{\phi(\beta(t), \varepsilon)}{\beta(t + \varepsilon)}.$$

It means that I need at least  $\frac{\phi(\beta(t), \varepsilon)}{\beta(t + \varepsilon)}$  symbols to characterize the interval  $\mathcal{R}$ . With 1 bit, we can characterize at most two symbols which are '0' and '1'. With 2 bits, we can characterize at most four symbols, which are '00', '01', '10', '11'. So, with  $B$  bits, we can characterize at most  $2^B$  symbols. Since there are  $N$  symbols, we need at least  $\log_2 N$  bits to characterize all of them. As  $N \geq \frac{\phi(\beta(t), \varepsilon)}{\beta(t + \varepsilon)}$ , at least  $\log_2 \left( \frac{\phi(\beta(t), \varepsilon)}{\beta(t + \varepsilon)} \right)$  bits are needed, and the average bit-rate during this interval is

$$r(t, t + \varepsilon) \geq \frac{\log_2 \left( \frac{\phi(\beta(t), \varepsilon)}{\beta(t + \varepsilon)} \right)}{\varepsilon}.$$

To pursue the instantaneous bit-rate at time  $t$ , we let  $\varepsilon$  to be infinitely close to 0, i.e.

$$r(t) = \lim_{\varepsilon \rightarrow 0} r(t, t + \varepsilon) \geq \frac{1}{\ln 2} \left( \frac{f(\beta(t))}{\beta(t)} + \alpha \right).$$

Since the bit-rate is always non-negative, we have equation (10).  $\blacksquare$

**Remark 4.3:** The term  $\frac{f(\beta(t))}{\beta(t)}$  in equation (10) indicates that the fastest increasing speed of the system. Our necessary stabilizing bit-rate is proportional to this speed, because we have to use the upper bound of the state to do the quantization.

Theorem 4.2 is then applied to linear systems, and we have the following corollary.

**Corollary 4.4:** If the plant (1) is linear, then the necessary bit-rate  $r_s$  to maintain the performance level (5) in safe regime is  $\frac{L + \alpha}{\ln 2}$ , i.e.

$$r_s = \frac{L + \alpha}{\ln 2}. \quad (13)$$

*Remark 4.5:* The necessary bit-rate of continuous system given by Hespanha et.al in [9] is recovered by our result in equation (14) with  $\alpha = 0$ , since Hespanha et.al only considered uniform boundness.

After talking about the necessary bit-rate in safe regime, we will construct an event triggered encoding and decoding algorithm to provide a sufficient bit-rate. The goal is to achieve the necessary bit-rates after the initial packet(indicating regime changing) is transmitted, at least, for linear systems.

### B. Sufficient Bit-Rate in Safe Regime

Let's first propose an event triggered encoding and decoding algorithm in safe regime, and then analyze the bit-rate of this algorithm. Finally, we would like to discuss how to choose the parameters to achieve the necessary bit-rates in finite time for linear systems.

Before giving the algorithm, we would like to introduce some variables to make the algorithm easy to read. Let  $M_e$  and  $M_d$ , stored in encoder and decoder respectively, indicate which regime the system is running in. To be more specific,

$$M_e, M_d = \begin{cases} 0, & \text{in safe regime;} \\ 1, & \text{in unsafe regime,} \end{cases}$$

The basic idea of the following algorithm is that since both encoder and decoder agree with a triggering event. So, once transmission occurs, the magnitude of the state is fixed. So we only need 1 bit to indicate the state is positive or negative. The details of this algorithm are given below.

**Algorithm 4.6: Event triggered encoding and decoding in safe regime.**

#### Encoder algorithm

If system enters safe regime, i.e.  $M_e \neq 0$  and  $|x(t)| \leq x_a$ , then

##### 1) Initialization

- Update system regime to be safe, i.e.  $M_e = 0$ .
- Reset timer to 0, and start the timer.
- Quantize the state  $x(0)$  by setting

$$q = \begin{cases} 00, & \text{if } x(0) \geq 0; \\ 01, & \text{if } x(0) < 0. \end{cases}$$

- Send  $q_0$  to decoder.
- Wait for the acknowledgement (with neglectable delay) from decoder or a state jump of  $-\frac{x_a}{2}$  if  $q = 00$ , or  $\frac{x_a}{2}$  if  $q = 01$ .
- Record the delay of  $q_0$  as  $\Delta_0$ .

##### 2) If $|x(t)| = \theta(x_a, t - \Delta_0)$ , where

$$\theta(x_a, t - \Delta_0) = x_a e^{-\alpha(t - \Delta_0)},$$

then

- Quantize the state  $x(t)$  as

$$q = \begin{cases} 0, & \text{if } x(t) \geq 0; \\ 1, & \text{if } x(t) < 0. \end{cases}$$

- Send  $q$  to decoder.

##### 3) If system enters unsafe regime, go to encoder algorithm in unsafe regime. Otherwise, go to step 2).

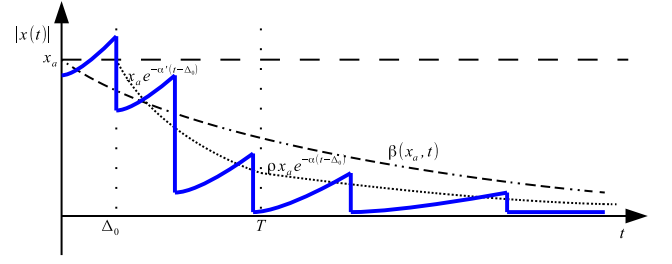


Fig. 3. State trajectory in safe region

#### Decoder algorithm

If system enters safe regime, i.e.  $q = 00$  or  $01$ , then

##### 1) Initialization

- Update system regime to be safe, i.e.  $M_d = 0$ .
- Reset timer to be 0, and start the timer.
- Initialize state estimate as

$$\hat{x}(0) = \begin{cases} \frac{x_a}{2}, & \text{if } q = 00; \\ -\frac{x_a}{2}, & \text{if } q = 01. \end{cases}$$

- Send an acknowledgement to encoder.

##### 2) Wait for the quantized data from encoder.

##### 3) If system enters unsafe regime, i.e. $q = 10$ or $11$ , then go to decoder algorithm in unsafe regime. Otherwise, go to step 3).

##### 4) Update state estimate as

$$\hat{x}(t) = \begin{cases} \theta(x_a, t), & \text{if } q = 0; \\ -\theta(x_a, t), & \text{if } q = 1. \end{cases}$$

##### 5) Go to step 2).

Given the encoding and decoding algorithm, now let's analyze the sufficient bit-rate to guarantee that the system performance is preserved. Since the number of bits transmitted at each time is fixed (2 bits for regime changing and 1 bit if regime doesn't change), to analyze the bit-rate is the same to analyze the acceptable delay of each packet.

Figure 3 shows a typical behavior of the system in safe regime. Basically, the safe regime can be divided into 2 time horizons. The first time horizon  $[0, \Delta_0)$  is the one during which the 'regime changing' packet is transmitted. The second time horizon  $[\Delta_0, t_f)$ , where  $t_f$  indicates the terminal time of this safe regime, is the time horizon when we use threshold function  $\theta$  to trigger transmission. Now, let's study the acceptable delay in each time horizon, and then further discuss how to choose parameter  $\rho$  in performance level (5). By 'acceptable delay', we mean that with 'acceptable delay', the system still satisfies the performance level (5).

For time horizon  $[0, \Delta_0]$ , we need to guarantee that the performance level (5) is satisfied and the state trajectory comes back to the absorbing region after  $q_0$  is received and the control is applied, i.e.  $|x(\Delta_0)| \leq x_a$ . The next lemma gives an upper bound on the initial delay  $\Delta_0$ .

*Lemma 4.7:* If

$$\Delta_0 \leq \min \left\{ \frac{\ln(\frac{3}{2})}{L}, \frac{\ln \rho}{L + \alpha} \right\}, \quad (14)$$

then  $|x(\Delta_0)| \leq x_a$ .

*Proof:* Let  $x_0 = x(0)$  indicate the initial state when the state enters the safe regime. Since the plant (1) is locally Lipschitz with Lipschitz constant  $L$ , we have

$$\begin{aligned} |x(\Delta_0)| &= \left| |\phi(x_0, \Delta_0)| - \frac{x_a}{2} \right| \\ &\leq \left| |x_0|e^{L\Delta_0} - \frac{x_a}{2} \right| \end{aligned}$$

To make sure  $|x(\Delta_0)| \leq x_a$ , we force

$$\left| |x_0|e^{L\Delta_0} - \frac{x_a}{2} \right| \leq x_a.$$

The solution of the inequality above is

$$\Delta_0 \leq \frac{\ln\left(\frac{3x_a}{2|x_0|}\right)}{L}.$$

We require that the inequality holds for any  $x_0 \in [-x_a, x_a]$ , so we have the first term of equation (15).

Besides, we also need to guarantee that the performance level is satisfied, i.e.  $|x(t)| \leq \rho x_a e^{-\alpha t}$  for all  $t \in [0, \Delta_0]$ . To guarantee this, we force

$$|x_0|e^{L\Delta_0} \leq \rho x_a e^{-\alpha \Delta_0}.$$

The solution of the equation above is

$$\Delta_0 \leq \frac{\ln\left(\frac{\rho x_a}{|x_0|}\right)}{L + \alpha}.$$

We require that the inequality holds for any  $|x_0| \leq x_a$ , so we have the second term of equation (15). ■

With the upper bound on the initial delay given by Lemma 4.7, we assure that the system state is bounded by the threshold  $\theta$  at time  $\Delta_0$ . Suppose the transmission occurs during the second time horizon, i.e.  $[\Delta_0, t_f]$ . We require not only that the state is bounded by the threshold function after packet is received and the control input is applied, i.e.  $|x(r_i)| \leq \theta(x_a, r_i - \Delta_0)$ , but also that the performance level indicated by equation (5) is guaranteed, i.e.  $|x(t)| \leq \beta(x_a, t)$  for all  $t \in [s_i, r_i]$ . To do that, we provide an upper bound on the delay  $\Delta_i$  in the next lemma.

*Lemma 4.8:* Suppose the  $i$ th transmission occurs during the time interval  $[\Delta_0, t_f]$ , i.e.  $|x(s_i)| = x_a e^{-\alpha(t-\Delta_0)}$ . If

$$\Delta_i \leq \min \left\{ \frac{\ln 2}{L + \alpha}, \frac{\ln(\rho e^{-\alpha \Delta_0})}{L + \alpha} \right\}, \quad (15)$$

where  $\rho$  is chosen such that  $\rho e^{-\alpha \Delta_0} > 1$  for any  $\Delta_0$  satisfies (15), then  $|x(r_i)| \leq \theta(x_a, r_i - \Delta_0)$  and  $|x(t)| \leq \beta(x_a, t)$  for all  $t \in [s_i, r_i]$ .

*Proof:* Use the same technique as in the proof of Lemma 4.7, we have

$$|x(r_i)| \leq |x(s_i)|e^{L\Delta_i} - x_a e^{-\alpha(s_i - \Delta_0 + \Delta_i)}.$$

To make sure  $|x(r_i)| \leq \theta(x_a, r_i - \Delta_0)$ , we force

$$|x(s_i)|e^{L\Delta_i} - x_a e^{-\alpha(s_i - \Delta_0 + \Delta_i)} \leq x_a e^{-\alpha(s_i - \Delta_0 + \Delta_i)}.$$

Realizing that  $|x(s_i)| = x_a e^{-\alpha(s_i - \Delta_0)}$ , we get

$$\Delta_i \leq \frac{\ln 2}{L + \alpha}.$$

To guarantee  $|x(t)| \leq \rho x_a e^{-\alpha t}$  for all  $t \in [s_i, r_i]$ , we first notice that  $|x(t)| \leq |x(s_i)|e^{L\Delta_i}$  for all  $t \in [s_i, r_i]$ . If

$$|x(s_i)|e^{L\Delta_i} \leq \rho x_a e^{-\alpha(s_i + \Delta_i)},$$

then  $|x(t)| \leq \beta(x_a, t)$ . Solving the equation above with  $|x(s_i)| = x_a e^{-\alpha(s_i - \Delta_0)}$ , we have

$$\Delta_i \leq \frac{\ln(\rho e^{-\alpha \Delta_0})}{L + \alpha}.$$

*Remark 4.9:* The requirement in  $\rho$ , i.e.  $\rho e^{-\alpha \Delta_0} > 1$  indicates that after time  $\Delta_0$ , the threshold function is always below the performance level.

From Lemma 4.8, we see that if we properly choose parameter  $\rho$ , we can achieve the maximum of the upper bound on delay in the second time horizon.

*Corollary 4.10:* Suppose the  $i$ th transmission occurs during the time interval  $[T, t_f]$ . If we choose  $\rho$  such that

$$\rho \leq 2^{\frac{\alpha+L}{L}},$$

then as long as the delay satisfies

$$\Delta_i \leq \frac{\ln 2}{L + \alpha},$$

$|x(r_i)| \leq \theta(x_a, r_i - \Delta_0)$  and  $|x(t)| \leq \beta(x_a, t)$  for all  $t \in [s_i, r_i]$ .

*Proof:* If the following inequality holds, then this corollary is true.

$$\frac{\ln 2}{L + \alpha} \leq \frac{\ln(\rho e^{-\alpha \Delta_0})}{L + \alpha}.$$

Solve the equation above we have

$$\Delta_0 \leq \frac{\ln \frac{\rho}{2}}{\alpha}.$$

To make sure that the initial delay in (15) still works, we force

$$\frac{\ln \frac{\rho}{2}}{\alpha} \geq \frac{\ln \rho}{L + \alpha}.$$

Solve this equation, we have  $\rho \leq 2^{\frac{\alpha+L}{L}}$ . ■

To conclude Lemma 4.7 and corollary 4.10, we have the theorem about the sufficient bit-rate in safe regime.

*Theorem 4.11:* In safe regime, if  $q_0$  is transmitted with bit-rate  $r(0)$  satisfying

$$r(0) \geq \bar{r}_s(0) = \frac{2}{\min \left\{ \frac{\ln(\frac{3}{2})}{L}, \frac{\ln \rho}{L + \alpha} \right\}},$$

and  $q_i$  for  $i = 1, 2, \dots$  is transmitted with bit-rate  $r(i)$  satisfying

$$r(i) \geq \bar{r}_s(i) = \frac{L + \alpha}{\ln 2},$$

then the performance level (5) can be guaranteed with  $\rho \geq 2^{\frac{\alpha+L}{L}}$ .

*Remark 4.12:* If the system is nonlinear, it's easy to verify that  $\underline{r}_s \leq \bar{r}_s$ . If the system is linear, the equality holds, i.e.  $\underline{r}_s = \bar{r}_s$ . Therefore, for linear system, we can say that after the initial delay, the performance level (5) is guaranteed with the necessary stabilizing bit-rate in safe regime.

## V. SUFFICIENT BIT-RATE IN UNSAFE REGIME

In this section, we first propose an algorithm for unsafe regime to assure the resilience, and then analyze the bit-rate of this algorithm.

The basic idea of our algorithm in unsafe regime is very simple. For each transmission, the encoder asks the decoder to move towards the origin by  $2x_a$ , since this is the maximum amount that the controller can apply. Once upon the packet is received and the control input is applied, the encoder checks the state again. If the state is still out of the absorbing region, i.e.  $|x(t)| \geq x_a$ , the encoder asks the decoder to move towards the origin by  $2x_a$  again. Otherwise, the system enters the safe regime, and uses the algorithm in safe regime instead.

**Algorithm 5.1: Event triggered encoding and decoding in unsafe regime**

### Encoder algorithm

If system enters unsafe regime, then

#### 1) Initialization

- Update system regime to be unsafe, i.e.  $M_e = 1$ .
- If  $|x(t)| \leq x_a$ , go to encoder algorithm in safe regime.
- Reset time to be 0.
- Quantize the state  $x(0)$  as

$$q = \begin{cases} 10, & \text{if } x(0) \geq 0; \\ 11, & \text{if } x(0) < 0. \end{cases}$$

- Send  $q$  to decoder.

- 2) Wait for the acknowledgement from decoder or a jump of magnitude  $2x_a$  towards the origin.
- 3) If  $|x(t)| \leq x_a$ , go to encoder algorithm in safe regime.
- 4) Quantize the state  $x(t)$  as

$$q = \begin{cases} 0, & \text{if } x(0) \geq 0; \\ 1, & \text{if } x(0) < 0. \end{cases}$$

- 5) Send  $q$  to decoder.
- 6) Go to step 2).

### Decoder algorithm

If the system enters the unsafe regime, i.e.  $q = 10$  or  $11$ , then

#### 1) Initialization.

- Update system regime to be unsafe, i.e.  $M_d = 1$ .
- Estimate state as

$$\hat{x}(0) = \begin{cases} 2x_a, & \text{if } q = 10; \\ -2x_a, & \text{if } q = 11. \end{cases}$$

- Send an acknowledgement to encoder.

- 2) Wait for quantized data from encoder.
- 3) If system enters safe regime, go to decoder algorithm in safe regime.
- 4) Estimate state as

$$\hat{x}(t) = \begin{cases} 2x_a, & \text{if } q = 0; \\ -2x_a, & \text{if } q = 1. \end{cases}$$

- 5) Send an acknowledgement to encoder.
- 6) Go to step 2).

*Remark 5.2:* Under unsafe regime, Brockett and Liberzon's zoom out idea in [10] can also be applied. However,

in this zoom out strategy, the controller is assumed to have unlimited power to drive the system state back to the neighborhood. This is not practical in most cases. Considering this limitation, we propose algorithm 5.1. With this algorithm, for each time, only  $2x_a$  amount of control is applied for each time, which is more practical to do.

Given the algorithm above, since the number of bits is fixed in each packet. To analyze the sufficient bit-rate of this algorithm is the same to analyze the upper bound on the delay of each packet which assures resilience.

*Lemma 5.3:* In unsafe region, if the delay of the  $j$ th transmission,  $\Delta_j$ , satisfies

$$\Delta_j < \frac{2x_a}{L} \quad (16)$$

then there exist a pair of encoder and decoder to guarantee that the system is resilient, i.e. for any  $|x(0)| > x_a$ , there exists a finite time  $\tau_f$  such that

$$|x(\tau_f)| \leq x_a.$$

*Proof:* If for every transmission, the magnitude of the state is decreased, i.e.

$$|x(s_{j+1})| < |x(s_j)|, \forall j = 0, 1, 2, \dots \quad (17)$$

then there must exist a constant  $\sigma_j \in (0, 1)$  such that

$$|x(s_{j+1})| \leq \sigma_j |x(s_j)|, \forall j = 0, 1, 2, \dots.$$

Hence, we have

$$|x(s_N)| = \prod_{j=0}^{N-1} \sigma_j |x(0)| \leq (\sigma^*)^N |x(0)|,$$

where  $\sigma^* = \max_{j=0,1,\dots,N-1} \sigma_j$ . Since  $\sigma_j \in (0, 1)$  for all  $j = 0, 1, \dots, N$ ,  $\sigma^* \in (0, 1)$ . So, for any  $x(0)$  and  $x_a$ , I can always find a finite integer  $N$  such that

$$|x(s_N)| \leq (\sigma^*)^N |x(0)| \leq x_a.$$

In other words, the system is resilient.

To make sure (18) holds, we first derive an upper bound on  $|x(s_{j+1})|$ , and then let this upper bound less than  $|x(s_j)|$  to get the longest acceptable delay.

We know that during interval  $[s_j, s_{j+1})$ , the derivative of  $|x|$  satisfies

$$\frac{d|x|}{dt} \leq \left| \frac{dx}{dt} \right| = |f(x)|.$$

Because it is assumed that the increasing speed of the state satisfies equation (4), we have

$$\frac{d|x|}{dt} \leq \bar{L}.$$

Therefore,

$$|x(s_{j+1})| \leq |x(s_j)| + \bar{L}\Delta_j - 2x_a.$$

To make sure that (18) is true, we let

$$|x(s_j)| + \bar{L}\Delta_j - 2x_a < |x(s_j)|,$$

which is the same to say that  $\Delta_j < \frac{2x_a}{\bar{L}}$ . ■

With Lemma 5.3, we give the sufficient bit-rate to guarantee resilience in unsafe regime in the next theorem.

*Theorem 5.4:* In unsafe regime, if  $q_0$  is transmitted with bit-rate

$$r(0) > \bar{r}_u(0) = \frac{\bar{L}}{x_a},$$

and the bit-rate of the  $j$ th transmission for  $j = 1, 2, \dots, N$ ,  $r(j)$ , satisfies

$$r(j) > \bar{r}_u(j) = \frac{\bar{L}}{2x_a},$$

where  $N$  indicates the last transmission index in unsafe regime, then there exists a pair of encoder and decoder to guarantee that the system is resilient, i.e. for any  $|x(0)| > x_a$ , there exists a finite time  $\tau_f$  such that

$$|x(\tau_f)| \leq x_a.$$

$\bar{r}_u$  is called the sufficient resilience bit-rate.

*Remark 5.5:* Notice that the sufficient resilience bit-rate  $\bar{r}_u$  is independent with the initial state  $x(0)$ . It means that no matter how far away the system state is from the origin, the state can always be pushed back into the absorbing region with the same bit-rate.

## VI. SIMULATION RESULTS

In this section, we first use a nonlinear case to demonstrate theorem 5.4 which says that the system is resilient against any state jump with a bit-rate that is only a little higher than the sufficient resilience bit-rate. Besides, we also show that a certain performance level is guaranteed under safe regime. A linear case is then presented to show that the system is still resilient and only necessary stabilizing bit-rate is used to guarantee performance level under safe regime.

Consider the following nonlinear system:

$$\dot{x} = \begin{cases} |x|\sin(x) + u + w, & \text{if } |x| \leq \rho x_a; \\ \arctan(x) + u + w, & \text{if } |x| > \rho x_a. \end{cases}$$

with  $-0.9x_a$  as the initial state.  $x_a = 0.5$  is the magnitude of absorbing region. We choose our performance level to be  $\beta(t) = \rho x_a e^{-0.2(t-t_0)}$  with  $\rho = 2^{\frac{0.2+\bar{L}}{L}}$ , where  $L = 1$  is the Lipschitz constant under safe regime.  $\bar{L} = \pi/2$  is the uniform bound of  $|f(x)$  when  $|x| > \rho x_a$ . When the system is under unsafe regime, we choose the maximum delay to be  $0.99 * 2 * x_s / \bar{L}$ .

The system is run for 75 seconds with maximum delay, and two impulsive disturbances hit the system with magnitude  $5x_a$  and  $15x_a$  at time 10s and 30s, respectively. The system performance is given in figure 4 with  $x$ -axis and  $y$ -axis indicating time and state, respectively. At time 8s and 50s, the state (solid line) jumps to 2.5 and 7.8, respectively. With the same maximum delay which is 0.99 of the maximum acceptable delay, we see from the plot that the system state comes back to the absorbing region in 5 seconds for the first attack and 34 seconds for the second attack, which shows that as long as the bit-rate is higher than the sufficient resilience bit-rate, no matter how far away

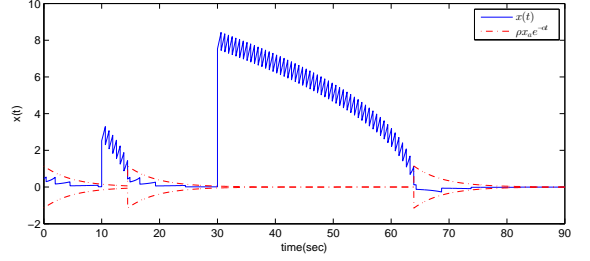


Fig. 4. System performance for the nonlinear system

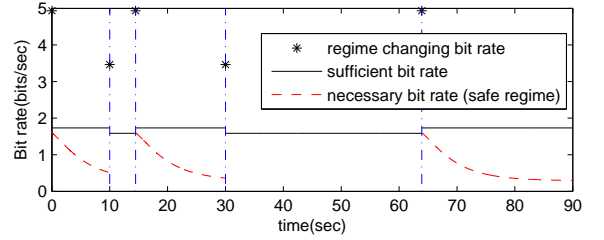


Fig. 5. Necessary and sufficient bit-rates for the nonlinear system

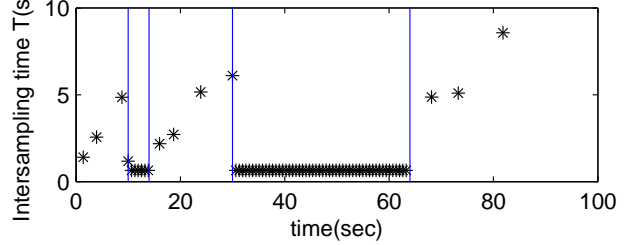


Fig. 6. Inter-sampling interval for the nonlinear system

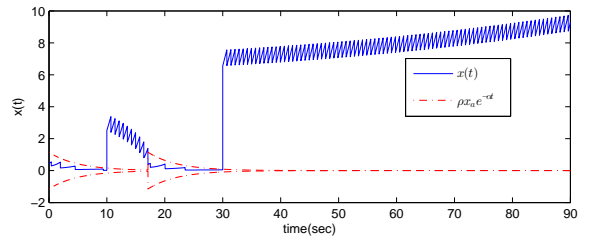


Fig. 7. System performance for nonlinear system with 0.9 of sufficient resilience bit-rate

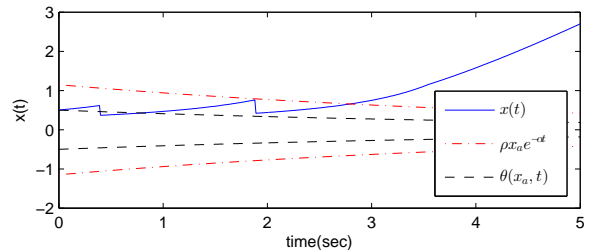


Fig. 8. System performance for nonlinear system with 0.9 of necessary stabilizing bit-rate



the state is, the system is resilient, i.e. comes back to the absorbing region in finite time. After the system comes back to the absorbing region, which are the time intervals  $[0, 8]$ ,  $[42, 50]$  and  $[58, 75]$ , the performance level (dot dashed line) is guaranteed.

Figure 5 gives the bit-rates of the system.  $x$ -axis is time, and  $y$ -axis is bit-rate. Since the regime changing packet has 2 bits, the corresponding bit-rate (stars) is higher than the other packets. We also notice that there is a gap between the the necessary stabilizing bit-rates (dashed line) and the sufficient stabilizing bit-rates (solid line) under safe regimes (intervals  $[0, 8]$ ,  $[42, 50]$  and  $[58, 75]$ ). That is because our sufficient stabilizing bit-rates are calculated from a global Lipschitz constant. This global Lipschitz constant becomes more conservative as the state goes to 0.

The inter-sampling interval  $s_{i+1} - s_i$  is given in figure 6 with  $x$ -axis indicating time and  $y$  axis indicating inter-sampling interval. In unsafe regimes, the inter-sampling intervals are constant since we always set out delay to be the maximum delay. In safe regimes, we can see that the inter-sampling intervals are increasing with respect to time, or in other words, increasing as  $x$  goes to the origin, which implies that our system is efficient attentive [11].

To test how tight our sufficient resilience bit-rate is, we use 0.9 of the sufficient resilience bit-rate to run the system when the system is under unsafe regime. Figure 7 gives the system performance with the same disturbances that we use in figure 4. We can see that for the first disturbance with magnitude of  $5x_a$ , the system is still resilient, but for the second disturbance with magnitude of  $15x_a$ , the system can't come back to the absorbing region again. That's because that when  $x = 5x_a$ ,  $\bar{L}$  doesn't bound  $|f(x)|$  tight enough, but when  $x = 15x_a$ ,  $\bar{L}$  is a very close upper bound on  $|f(x)|$ .

To test whether our necessary stabilizing bit-rate is true, we use 0.9 of this bit-rate to run the system under safe regime. Figure 8 gives the system performance with initial state to be  $x_a$ . we see that the state trajectory (solid line) becomes unbounded. That's because the delay is so long that after the control input is applied, the state is still above the threshold (dashed line), the event  $|x(t)| = \theta(x_a, t)$  will never occur, and there is no transmission anymore.

Now, let's consider a linear system as below.

$$\dot{x} = \begin{cases} 2x + u + w, & \text{if } |x| \leq \rho x_a; \\ 4 \arctan(x) + u + w, & \text{if } |x| > \rho x_a, \end{cases}$$

with initial state to be 2. The magnitude of absorbing region  $x_a$  is 3, and the performance level is chosen to be  $\beta(t) = \rho x_a e^{-\alpha}$  with  $\rho = 2^{\frac{1+\alpha}{2}}$ . When  $|x| > \rho x_a$ , the uniform bound on  $|f(x)|$  is  $\bar{L} = \pi/2$ . The maximum delay under unsafe regime is chosen to be  $0.1 \frac{2x_a}{\bar{L}}$ .

We ran the system for 15 seconds. Figure 9 shows the performance of the system.  $x$ -axis indicates time, and  $y$ -axis indicates state. The system is attacked at time 5s with the state (solid line) jumping to 30. We can see that the state comes back to the absorbing region in about 2 seconds, which demonstrates that our system is resilient. Under safe regimes, which are intervals  $[0, 5]$  and  $[7, 15]$ ,

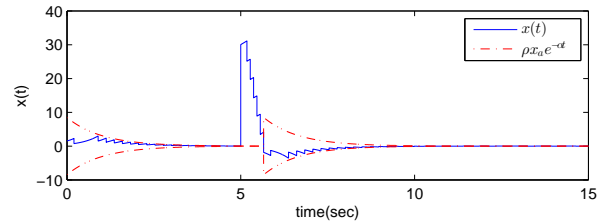


Fig. 9. System performance for the linear system

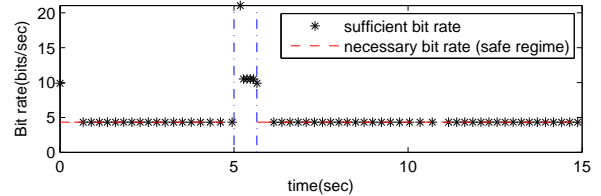


Fig. 10. Necessary and sufficient bit-rates for the linear system

the performance level (dashed line) is always guaranteed. The necessary and sufficient bit-rates are shown in figure 10 with  $x$ -axis to be the time, and  $y$ -axis indicating bit-rate. We notice that under safe regimes, the gap between the sufficient stabilizing bit-rate (stars) and the necessary bit-rate (dashed line) is 0 (except the first packet indicating 'regime change'). It demonstrates our assertion that for linear systems, we can use only necessary stabilizing bit-rate to guarantee the performance level.

## VII. CONCLUSIONS AND FUTURE WORKS

This paper studies the resilience of event triggered systems against impulsive disturbances with unknown magnitude. An even triggered strategy is provided to achieve the resilience, and the sufficient resilience bit-rate is derived. This sufficient resilience bit-rate is independent with the initial state, which means that no matter how far away the system state is driven to, we can always use the same bit-rate to move the state back to the absorbing region.

## VIII. ACKNOWLEDGMENTS

The authors acknowledge the partial financial support of the National Science Foundation NSF-CNS-0931195 and NSF-ECCS-0925229.

## REFERENCES

- [1] C. Rieger, D. Gertman, and M. McQueen, "Resilient control systems: next generation design research," in *Human System Interactions, 2009. HSI'09. 2nd Conference on*. IEEE, 2009, pp. 632–636.
- [2] Q. Zhu and T. Basar, "Robust and resilient control design for cyber-physical systems with an application to power systems," in *Proc. of 50th IEEE Conference on Decision and Control and European Control Conference, Orlando, Florida, 2011*.
- [3] K. Årzén, "A simple event-based PID controller," in *Proc. 14th IFAC World Congress*, vol. 18, 1999, pp. 423–428.
- [4] K. Astrom and B. Bernhardsson, "Comparison of Riemann and Lebesgue sampling for first order stochastic systems," in *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, vol. 2. IEEE, 2002, pp. 2011–2016.
- [5] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," in *Automatic Control, IEEE Transactions on*, vol. 52, no. 9. IEEE, 2007, pp. 1680–1685.



- [6] X. Wang and M. Lemmon, "Self-Triggered Feedback Control Systems With Finite-Gain  $L_2$  Stability," in *Automatic Control, IEEE Transactions on*, vol. 54, no. 3. IEEE, 2009, pp. 452–467.
- [7] W. Wong and R. Brockett, "Systems with finite communication bandwidth constraints. ii. stabilization with limited information feedback," *Automatic Control, IEEE Transactions on*, vol. 44, no. 5, pp. 1049–1053, 1999.
- [8] S. Tatikonda and S. Mitter, "Control under communication constraints," *Automatic Control, IEEE Transactions on*, vol. 49, no. 7, pp. 1056–1068, 2004.
- [9] J. Hespanha, A. Ortega, and L. Vasudevan, "Towards the control of linear systems with minimum bit-rate," in *Proc. 15th Int. Symp. on Mathematical Theory of Networks and Systems (MTNS)*. Citeseer, 2002.
- [10] R. Brockett and D. Liberzon, "Quantized feedback stabilization of linear systems," *Automatic Control, IEEE Transactions on*, vol. 45, no. 7, pp. 1279–1289, 2000.
- [11] X. Wang and M. Lemmon, "Attentively efficient controllers for event-triggered feedback systems," in *IEEE Conference on Decision and Control*, 2011.