

Precision on Demand: Propositional Logic for Event-Trigger Threshold Regulation

Valdemar Tang, Claudio Gomes and Daniel E. Lucani

Abstract—We introduce a novel event-trigger threshold (ETT) regulation mechanism based on the quantitative semantics of propositional logic (PL). We exploit the expressiveness of the PL vocabulary to deliver a precise and flexible specification of ETT regulation based on system requirements and properties. Additionally, we present a modified ETT regulation mechanism that provides formal guarantees for satisfaction/violation detection of arbitrary PL properties. To validate our proposed method, we consider a convoy of vehicles in an adaptive cruise control scenario. In this scenario, the PL operators are used to encode safety properties and the ETTs are regulated accordingly, e.g., if our safety metric is high there can be a higher ETT threshold, while a smaller threshold is used when the system is approaching unsafe conditions. Under ideal ETT regulation conditions in this safety scenario, we show that reductions between 41.8 - 96.3% in the number of triggered events is possible compared to using a constant ETT while maintaining similar safety conditions.

Index Terms—Propositional Logic, Event-triggering mechanisms, Interval Arithmetic

I. INTRODUCTION

Cyber-physical systems can collect large amounts of sensor data making them costly to operate over time. In such systems, data is often transmitted periodically. This causes a consistent overhead of operating the system even when no new or interesting events are happening. Freeing up communication resources from sensors that do not require high accuracy would also allow the system to increase communication with sensors measuring critical values that require high accuracy. The criticality associated with each sensor changes in time and will depend on specific system properties and requirements. Thus, there is a clear motivation to develop methods to match data traffic to the safety and performance requirements of cyber-physical systems to avoid unnecessary resource consumption, delay or congestion in the communication network.

Event-triggering mechanism: A popular mechanism to implement efficient communication in the state-of-the-art (SOTA) is event-triggered state estimation and control. The fundamental idea is to only sample and/or transmit information when necessary. The necessity is defined by an event-triggering condition that must be fulfilled for the system to make use of the measurement. This can help to significantly reduce communication costs compared to periodic sampling and transmission [1]. The event-triggering condition typically

uses an event-triggering threshold (ETT) which indicates when the difference between new and known information is large enough to be transmitted. Figure 1 visualizes the conceptual differences between a periodic transmission strategy (top), a static threshold (middle) and a safety-dependent threshold (bottom). Intuitively, smaller thresholds (and thus better accuracy) are beneficial in unsafe situations. A similar parallel can be drawn to performance, where underperforming systems require better accuracy and well-performing systems can potentially operate with less sensor accuracy.

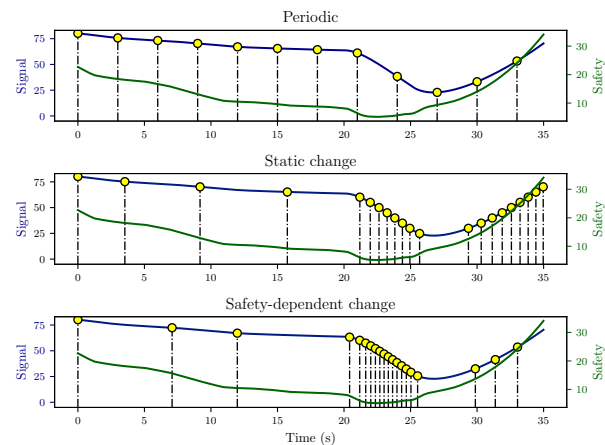


Fig. 1: Visualization of periodic transmission (top), static (middle) and safety-dependent (bottom) event-triggering conditions. Yellow dots indicate measurement transmission. The static event trigger prioritizes change in the signal regardless of safety, while the safety-dependent event trigger prioritizes change relative to the safety of the system.

Initially conceptualized in [2], [3], event-triggered techniques have since undergone significant development and have been applied to reduce communication in multi-agent consensus systems [4], networked control systems (NCSs) [5] and distributed state estimation [6] among others. As the goals of these systems are different, so are the event-triggering conditions. For example, in NCSs a central focus is on designing a controller and event-triggering mechanism to ensure stability of the controlled systems in the presence of transmission delays, dropouts and other phenomena introduced by the network. As a result, the event-triggering condition is often designed in combination with the controller and the system dynamics. Controller systems may employ state observers (a.k.a. state estimators), which means that using event-triggered techniques requires a corresponding adaptation

Manuscript created April, 2024; Revised August 2024; Valdemar Tang (valdemar.tang@ece.au.dk), Claudio Gomes and Daniel Lucani are with the Department of Electrical and Computer Engineering, Aarhus University, 8200 Aarhus, Denmark.

Copyright © 2024 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

of those observers. Several well-known techniques have been adapted to event-triggered state estimation, e.g., the Kalman filter [7], H_∞ filtering [8] and set-membership filtering [9].

Propositional Logic: While some properties such as stability and convergence are typically well-defined for a broad class of systems, other safety (and performance) properties require a formal definition on a per-system basis. Multiple formalisms can be used to achieve this goal. In this paper, we focus on signal-based Propositional Logic (PL) statements [10] as detailed later in Section II-D. PL is fundamentally different from stability analysis in that it provides a way to express and verify system properties based on time series data rather than exhaustively proving system stability. A PL formula outputs a verdict specifying whether the given property was satisfied or violated. Crucially, it provides a quantitative metric that signifies how much the property is violated or how far it is from being violated. Such properties can be related to performance (liveness) or safety.

Contribution: Since the design of the ETT depends on the goals of the system, and since PL provides a well-founded approach to describe and monitor such goals, we propose a method for leveraging the expressiveness of PL to enable precise regulation of the ETT based on system requirements. More specifically, we provide a parameterized ETT regulation mechanism for inequality PL properties and their negations and use the propositional operators and structure of the propositional property to refine the ETT. We provide a method for determining a set of parameters that under certain assumptions guarantee achieving the best possible accuracy before violating a safety or performance property. Additionally, we also provide guidance on parameter tuning for the proposed ETT regulation mechanism and evaluate our approach on a case study concerning a convoy of vehicles in an adaptive cruise control scenario. We compare the numerical results in terms of the number of triggered events for the time-triggered case, a constant ETT and our proposed ETT regulation mechanism. Finally, we explore key tradeoffs for different parameter configurations.

Structure: The considered system architecture, background terminology and problem formulation are presented in Section II followed by the proposed method in Section III. This method is applied to a simulated adaptive cruise control (ACC) scenario in Section IV. We discuss the results and limitations of our method in Section V as well as compare our work with SOTA. Finally, we present concluding remarks and planned future work in Section VI.

II. BACKGROUND

A. The event-triggered system architecture under study

We consider the system architecture in Fig. 2 with multiple smart sensors measuring a process. The true state vector of the system at time t_k is denoted by \mathbf{x}_{t_k} , the control input vector by \mathbf{u}_{t_k} and $f(\mathbf{x}_{t_k}, \mathbf{u}_{t_k})$ is the function that describes how the system evolves from $t_k \rightarrow t_{k+1}$. Sensor i measures a value y_{i,t_k} at time t_k given by $y_{i,t_k} = w_i(\mathbf{x}_{t_k})$. If the event-triggering condition is satisfied, the measurement y_{i,τ_{m+1}^i} is transmitted to the remote state estimator, where τ_{m+1}^i identifies the current event-triggering time instant for the

Notation	Description
$\hat{\mathbf{x}}_{t_k}, \hat{x}_{i,t_k}$	State estimate vector and state estimate vector element at time t_k .
$\mathbf{y}, \mathbf{y} \sim \varphi^P$	Measurable system output vector and signals to regulate ETTs for based on the inequality property φ^P respectively.
φ, φ^P	Arbitrary propositional and inequality property respectively
$\delta_{y_i, \varphi, t_{k+1}}(t_k, \epsilon_{y_i, \varphi})$	ETT of signal y_i at time t_k defined based on property φ .
$e(\cdot)$	Update error. Difference between predicted and current or last and current measurement.
$\rho(\varphi, \hat{\mathbf{x}}_{t_k})$	Robustness/satisfaction degree of the property φ and corresponding system states at time t_k .
$\Delta x, w(\Delta x)$	Interval and width of the variable x respectively.
$\Delta \hat{\mathbf{x}}_{t_{k+1} t_k}$	Vector containing intervals of predicted states at time t_{k+1} based on the states at time t_k .
$\underline{\rho}(\varphi, \Delta \hat{\mathbf{x}}_{t_{k+1} t_k}), \bar{\rho}(\varphi, \Delta \hat{\mathbf{x}}_{t_{k+1} t_k})$	Lower and upper bound respectively, of the predicted robustness interval of the inequality property φ at time t_{k+1} based on the state interval vector.
$\epsilon_{y_i, \varphi}, \lambda_{y_i, \varphi}, \epsilon_{\rho, \varphi^P}$	ETT regulation parameters related to signal y_i and/or inequality property φ^P .
$\zeta_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k})$	The normalized robustness of the property φ at time t_k . Variants with different super-scripts exist.

TABLE I: Overview of common notation used in the paper.

i th sensor. The state of the process is estimated by the remote state estimator using a model of the process, the previously estimated state $\hat{\mathbf{x}}_{t_{k-1}}$, measurements and ETT information from the sensors and monitor $\hat{\mathbf{y}}_{t_{k-1}}$ and control input $\mathbf{u}_{t_{k-1}}$. The system also has a set of properties based on the system requirements (Φ), which are monitored by the monitor. The monitor receives the $\hat{\mathbf{x}}_{t_k}$ and optionally \mathbf{u}_{t_k} to determine the state/satisfaction of the monitored properties. Based on the current state of the properties and the measurable system outputs \mathbf{y} , the monitor calculates suitable ETTs for each smart sensor. The smart sensor sampling and control system update frequency are assumed to be the same and synchronized. Additionally, we consider the case where the system, under the periodic transmission case (i.e. all samples are transmitted) satisfies all system properties.

B. ETT regulation strategies

The literature distinguishes between static and dynamic ETTs. **Static ETTs** depend on the current state estimate vector, are time-dependent or are constant, whereas **dynamic ETTs** have additional dynamic variables. We focus on state-dependent static ETTs and thus do not detail dynamic ETTs further and interested readers are referred to [11], [12], [13], [6], [14]. A general event-triggered transmission mechanism

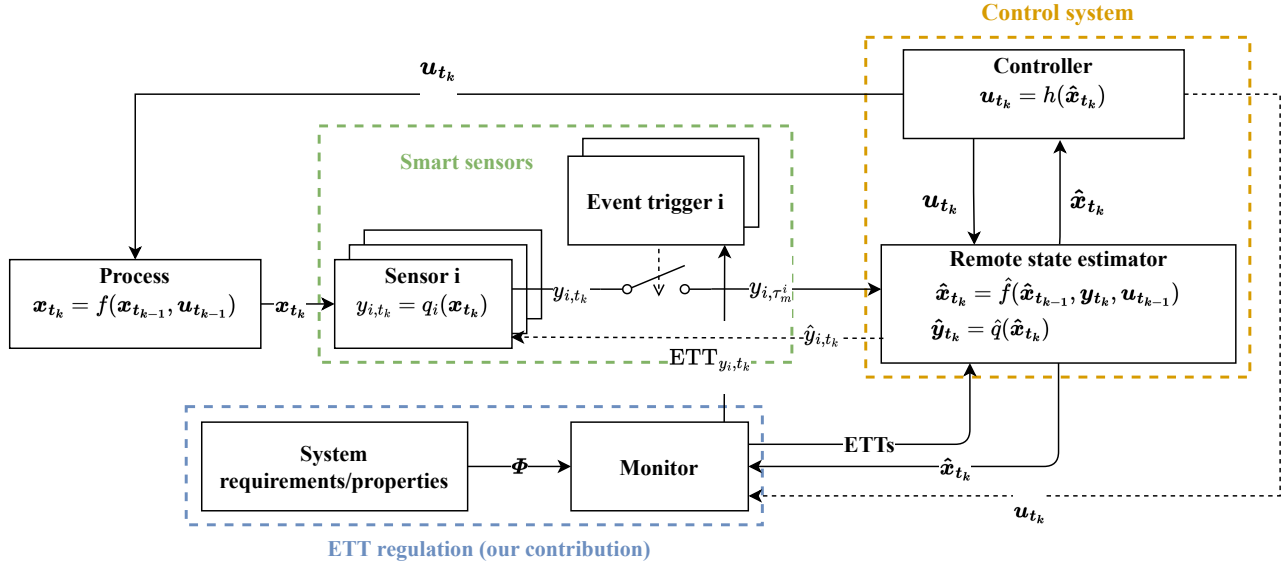


Fig. 2: An overview of the system architecture considered in this paper. Dotted black lines indicate an optional data flow. The control input u_{t_k} is necessary for the monitor if we wish to verify or monitor control actions in the system properties and \hat{y}_{i,t_k} is necessary if the innovation update error is used (described in Section II-B).

for a state-dependent static ETT can be defined as a sequence of event-triggering times [12] for a sensor i :

$$\tau_{m+1}^i = \inf\{t > \tau_m^i \mid g_e(e(\cdot), t, \dots) > g_x(\hat{x}_t, \dots)\}, \quad (1)$$

where $e(\cdot)$ is the update error and can be defined as the difference between the current measurement and previous transmitted measurement ($y_{i,t_k} - y_{i,\tau_m^i}$) (also known as the **Send-on-Delta (SOD)** method [15]) or the difference between the predicted and actual measurement

$$|\hat{y}_{i,t_k} - y_{i,t_k}| \quad (2)$$

also known as the **innovation** [1]. In the SOTA, state-dependent static ETTs are typically given by $g_e(e(\cdot), \Gamma^{\frac{1}{2}}) = \|\Gamma^{\frac{1}{2}}e(\cdot)\|^2$ and $g_x(\hat{x}_{t_k}, \Gamma^{\frac{1}{2}}) = \|\Gamma^{\frac{1}{2}}Z(\hat{x}_{t_k}, \dots)\|^2$ where Γ is a weighting matrix and $Z(\hat{x}_{t_k}, \dots)$ is a suitable function to be determined [11].

Example 1 (SOD Event-trigger). Consider a vehicle in an adaptive cruise control scenario that has an estimate of its position x^c as well as the preceding vehicle x^{c+1} where $x^c, x^{c+1} \in \mathbb{R}$. Intuitively, the further away vehicle c is from vehicle $c+1$, the more uncertainty we can allow. For example, we can define $Z(x^c, x^{c+1}) = \max(x^{c+1} - x^c - 10, 1)$, where the minimum ETT is then obtained once the vehicles are less than 11 meters apart. Now consider the position estimates $\hat{x}_{t_k}^c = 10$, $\hat{x}_{t_k}^{c+1} = 22$, the previous measured distance $y_{\tau_m^c} = 12$ from vehicle c to $c+1$ at time τ_m^c , $\sigma = 1$, and the weight $\Gamma^{\frac{1}{2}} = 0.2$. We calculate $Z(\cdot) = 22 - 10 = 12$. A new data sample is now taken at the sensor with the value of $y_{t_k} = 15$. The update error becomes: $e(y_{t_k}, y_{\tau_m^c}) = 15 - 12 = 3$. We

can now determine if an event is triggered by calculating the event-triggering condition:

$$g_e(e(y_{t_k}, y_{\tau_m^c}), \Gamma^{\frac{1}{2}}) = 0.2 \cdot 3 = 0.6,$$

$$g_x(Z(\hat{x}_{t_k}^c, \hat{x}_{t_k}^{c+1}), \Gamma^{\frac{1}{2}}) = 1 \cdot 0.2 \cdot 2 = 0.4.$$

As $g_e(\cdot) > g_x(\cdot)$, the event-triggering condition is satisfied and the measurement is transmitted.

A central point in the above example is that the ETT is relative to some function of \hat{x}_{t_k} . Thus, $g_x(\hat{x}_{t_k}, \dots)$ should be designed such that it produces a small value in situations that require a small ETT.

Event-triggering conditions can be designed for many different purposes. For example, event-triggered techniques have seen widespread use in multi-agent systems (MASs) [4]. In such systems, a key design objective could be to achieve consensus in bounded time, to which time-dependent ETTs have been implemented to reduce the ETT over time as the agents gradually achieve consensus [16]. Event-triggering techniques have since been developed for a wide range of system classes and setups from linear MASs [17] to heterogeneous nonlinear MASs [18]. Event-triggering mechanisms may also adapt to the state of the network. For example, if the network is congested, we may want to increase the ETT resulting in fewer triggered events to avoid further congestion, e.g., in [19]. Alternatively, the event-triggering condition may be designed in combination with a controller to ensure the stability of the controlled system, which is the primary focus in event-triggered networked control systems [20].

C. Event-triggered state estimation (ETSE)

The event-triggered transmission scheme requires an adaptation of the classical state estimation techniques for periodic

sampling and transmission. This is because the estimator has to handle both point (explicit) and set-valued (implicit) measurements where the latter is the result of the ETT [21]. There are two overall approaches for tackling this problem: a deterministic and a stochastic approach, each of which determines how the signals are modeled. A central issue in the stochastic approaches is how to deal with the non-gaussian distribution introduced by the ETT. Therefore, the traditional Kalman filter is not directly applicable. An early attempt was to implement the Kalman filter with intermittent observations [22], where the update step of incorporating the measurement is simply skipped at time instants when measurements are not available. This can however lead to sub-optimal performance compared to approaches that take the knowledge of intermittent observations into account [1]. Exact or optimal solutions to the event-based filtering problem are available but they require numerically estimating the probability distributions which makes them infeasible to use during runtime [21]. Therefore approximate solutions are necessary. For simplicity, we focus on state estimators for discrete-time linear systems of the type:

$$\begin{aligned} \mathbf{x}_{t_k+1} &= A\mathbf{x}_{t_k} + B\mathbf{u}_{t_k} + \mathbf{w}_{t_k}, \\ \mathbf{y}_{t_k} &= C\mathbf{x}_{t_k} + \mathbf{r}_{t_k}, \end{aligned} \quad (3)$$

where $\mathbf{x}_{t_k} \in \mathbb{R}^n$ is the system state vector at time t_k , $\mathbf{y}_{t_k} \in \mathbb{R}^m$ is the measurable system output at time t_k , $\mathbf{r}_{t_k} \in \mathbb{R}^m$ and $\mathbf{w}_{t_k} \in \mathbb{R}^n$ are the measurement and process noise respectively at time t_k and A, B and C are matrices of appropriate sizes.

Later in our experiments, we use an event-triggered state estimation method that explicitly utilizes the knowledge of the ETT to update the state estimate. One such method was proposed in [23] and was later compared with several other ETSE methods in [1] where it performed the best overall in terms of state estimation accuracy and resulting communication rate. We compared the method from [23] with an adaptation of the Kalman filter originally developed for the SOD method in [24] by empirically evaluating the estimation accuracy for the system model, where we found that they produced similar estimation accuracy for the case study scenario. As the modified Kalman filter (KF) from [24] is significantly simpler, we use this approach for our experiments.

Definition 1 (Modified Kalman filter adapted from [24]). *A modified Kalman filter which utilizes the knowledge of the event-triggering threshold is given by the standard Kalman prediction equations:*

$$\hat{\mathbf{x}}_{t_k}^- = A\hat{\mathbf{x}}_{t_{k-1}} + B\mathbf{u}_{t_{k-1}}, \quad (4)$$

$$P_{t_k}^- = AP_{t_{k-1}}A^T + Q, \quad (5)$$

a modification of the measurement and measurement covariance matrix R :

$$\bar{R}_{t_k} = R, \quad (6)$$

then for all signals if measurement of signal y_i is received at time t_k :

$$y_{i,t_k} = y_{i,t_k} \quad (7)$$

else use the predicted measurement $\hat{y}_{i,t_k}^- = C\hat{\mathbf{x}}_{t_k}^-$:

$$y_{i,t_k} = \hat{y}_{i,t_k}^-. \quad (8)$$

Then, if a measurement was not received for signal y_i , the value of a given measurement is assumed to be equal to the predicted value and have a uniform distribution with variance $\delta_{y_i,t_k}^2/3$ where δ_{y_i,t_k} is the ETT of signal y_i at time t_k which is then added to the measurement noise covariance matrix at the corresponding entry.

$$R_{t_k}(i, i) = R_{t_k}(i, i) + \delta_{y_i,t_k}^2/3. \quad (9)$$

We then proceed with the standard Kalman filter update equations:

$$K_{t_k} = P_{t_k}^- C^T (C P_{t_k}^- C^T + \bar{R}_{t_k})^{-1}, \quad (10)$$

$$\hat{\mathbf{x}}_{t_k} = \hat{\mathbf{x}}_{t_k}^- + K_{t_k}(\mathbf{y}_{t_k} - C\hat{\mathbf{x}}_{t_k}^-), \quad (11)$$

$$P_{t_k} = (I - K_{t_k}C)P_{t_k}^-, \quad (12)$$

where R is the measurement noise covariance matrix, Q is the process noise covariance matrix and A, B and C are the state-space equations from the system model in Eq. 3.

D. Propositional Logic

We consider signal-based PL statements with quantified semantics. PL is defined by the following recursive syntax

$$\varphi ::= p(\hat{\mathbf{x}}) > c \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2, \quad (13)$$

where $\hat{\mathbf{x}}$ is the state estimate vector, c is a constant, \neg is the negation operator \wedge, \vee are the logical *and* and *or* operators respectively. The syntax in Eq. (13) can be combined to form the well-known implication operator ($\varphi_1 \rightarrow \varphi_2 \equiv \neg\varphi_1 \vee \varphi_2$) which indicates that φ_1 implies φ_2 . We introduce the notation φ^p to refer to an inequality property as this will become useful later.

We adopt the quantitative metrics (robustness) of propositional properties from Signal Temporal Logic [25] that indicate how robustly a given property is violated/satisfied.

Definition 2 (Robustness adapted from [25]). *The robustness ρ of a PL property φ based on the system state $\hat{\mathbf{x}}_{t_k}$ at time t_k , denoted as $\rho(\varphi, \hat{\mathbf{x}}_{t_k})$ is defined recursively by:*

$$\rho(p(\hat{\mathbf{x}}_{t_k}) > c) = p(\hat{\mathbf{x}}_{t_k}) - c$$

$$\rho(\neg\varphi, \hat{\mathbf{x}}_{t_k}) = -\rho(\varphi, \hat{\mathbf{x}}_{t_k})$$

$$\rho(\varphi_1 \wedge \varphi_2, \hat{\mathbf{x}}_{t_k}) = \min(\rho(\varphi_1, \hat{\mathbf{x}}_{t_k}), \rho(\varphi_2, \hat{\mathbf{x}}_{t_k}))$$

$$\rho(\varphi_1 \vee \varphi_2, \hat{\mathbf{x}}_{t_k}) = \max(\rho(\varphi_1, \hat{\mathbf{x}}_{t_k}), \rho(\varphi_2, \hat{\mathbf{x}}_{t_k}))$$

E. Interval arithmetic

Due to the ETT, the values of the system outputs can be expressed as an interval $Y = [\underline{Y}, \bar{Y}]$ in terms of the ETT as [26]:

$$Y(y, \delta_y) = [y - \delta_y, y + \delta_y] = [\underline{Y}, \bar{Y}], \quad (14)$$

where δ_y is the ETT of the signal y and y is the previously transmitted value or the estimated value in the case of the SOD update error and innovation-based update error respectively.

For the remainder of the paper, we use the notation $\Delta x(\cdot)$ to refer to the interval of $x \in \mathbb{R}$ and $\Delta \mathbf{x}(\cdot)$ to refer to a vector containing intervals of $\mathbf{x} \in \mathbb{R}^n$. We say that an interval is degenerate if $\underline{Y} = \overline{Y}$, which we assume to be the case when an event is triggered as the true value of the signal is known. Furthermore, we introduce the addition of intervals X and Y as [26]: $X + Y = [\underline{X} + \underline{Y}, \overline{X} + \overline{Y}]$, the subtraction of intervals X and Y as: $X - Y = [\underline{X} - \overline{Y}, \overline{X} - \underline{Y}]$, and the width of an interval Y as:

$$w(Y) = \overline{Y} - \underline{Y} \quad (15)$$

and the product of two intervals X and Y as

$$X \cdot Y = [\min S, \max S] \text{ where } S = \{\underline{X}\underline{Y}, \underline{X}\overline{Y}, \overline{X}\underline{Y}, \overline{X}\overline{Y}\}. \quad (16)$$

We now present two results that will be useful later to provide property satisfaction detection guarantees.

Lemma 1. *Multiplying an interval $X = [\underline{X}, \overline{X}]$ where $\underline{X} \neq \overline{X}$ with a scalar $\alpha < 0$, reverses the interval s.t. $\alpha X = [\overline{X}\alpha, \underline{X}\alpha]$.*

Proof. See appendix A. □

Lemma 2. *The addition or subtraction of two intervals $X(x, \delta_x)$ and $Y(y, \delta_y)$ scaled by scalars α_X and α_Y respectively, where $\alpha_X \in \mathbb{R}, \alpha_Y \in \mathbb{R}$, produces an interval where $w(\alpha_X X(x, \delta_x) \pm \alpha_Y Y(y, \delta_y)) = 2|\alpha_X|\delta_x + 2|\alpha_Y|\delta_y$.*

Proof. See appendix B. □

For more information on interval arithmetic, the reader is referred to [26].

F. Problem formulation

We introduce the notation $\mathbf{y}_{\sim\varphi}$ where $\mathbf{y}_{\sim\varphi} \subseteq \mathbf{y}$ to denote the system outputs to which the ETT regulation should be applied based on the property φ . The set $\mathbf{y}_{\sim\varphi}$ is to be defined by the system designer.

We consider the problem of finding an event-triggering mechanism $ET(\Phi, \hat{\mathbf{x}}, \Xi)$ with a corresponding set of parameters Ξ to ensure the satisfaction of all system properties Φ while transmitting as few measurements as possible. This is stated formally as the following optimization problem:

$$\begin{aligned} \underset{ET(\Phi, \hat{\mathbf{x}}, \Xi)}{\operatorname{argmin}} \quad & \left(\sum_{y_i \in \mathbf{y}} |\{y_i, \tau_1^i, \dots, y_i, \tau_m^i\}| \right) \\ \text{s.t.} \quad & \forall \varphi \in \Phi \quad \forall t_k \in [t_0, t_{end}] \quad \rho(\varphi, \hat{\mathbf{x}}_{t_k}) > 0, \end{aligned} \quad (17)$$

Additionally, we assume that the original system under the time-triggered transmission approach (TT) satisfies all system properties. Thus we know that

$$\begin{aligned} ET(\cdot) = TT(T_s) \rightarrow \\ (\rho(\varphi, \hat{\mathbf{x}}_{t_k}) > 0) \quad \forall t_k \in [t_0, t_{end}] \quad \forall \varphi \in \Phi, \end{aligned} \quad (18)$$

where T_s denotes the sampling, transmission and controller update interval.

III. PL ROBUSTNESS-BASED ETT REGULATION

An ETT based on state information requires the designer to model $g_x(\hat{\mathbf{x}}_t, \dots)$ from Eq. (1) so that it outputs a small value in situations with a low desired state estimate uncertainty and vice versa. The robustness output of inequality PL properties provides exactly this. For example, we may allow a larger state-estimate uncertainty (and thus save communication resources) in definitely safe or well-performing states (large robustness) and require a lower uncertainty in unsafe/underperforming states (low robustness) as depicted in Fig. 1.

Relation with State of the Art (SOTA): The robustness of inequality PL properties and their negations in Definition 2 is calculated by subtracting some state-dependent value from a constant. This is similar to several implementations of $g_x(\hat{\mathbf{x}}, \dots)$ in SOTA. For example, [19] proposes a triggering mechanism that depends on the difference between the steady-state desired distance and the current distance between automatically controlled vehicles in a cooperative cruise-control scenario, which is similar to Example 1.

In [27], the authors study a multi-agent system that attempts to achieve consensus where the triggering condition is relative to a weighted difference of state estimates between agents. The smaller the difference, the smaller the ETT. This is analogous to a PL liveness property where we want to verify that the system achieves consensus. This similarity indicates that the quantitative semantics of PL properties is a suitable metric to base our ETT regulation mechanism on.

Section overview: Initially we define the baseline constant ETT policy in Section III-A. Then, in the following sections, we gradually build and extend our ETT regulation mechanism from inequality properties in Section III-B to include PL properties with inequality sub-properties in Section III-E and then arbitrary PL properties in Section III-F. Along the way, we outline the parameter selection procedure in Section III-C and provide guarantees for detecting property satisfaction first for inequality properties in Section III-D which we then extend to arbitrary PL properties in Section III-G.

A. Baseline scheme: Constant ETTs (CETT)

We consider the constant ETT policy CETT as a baseline to compare our proposed method to. We adapt Eq. (17) where we consider the problem of finding a set of constant ETTs that minimize the total number of triggered events, while still ensuring that the system satisfies all properties. Thus under the CETT scheme $\Xi = \{\delta_{y_0}, \dots, \delta_{y_i}\}$ and we add the constraint $\forall y_i \in \mathbf{y} \quad \delta_{y_i} \geq 0$. The constant ETTs are chosen such that they can ensure property satisfaction under the worst possible system conditions, and thus do not take advantage of less strict accuracy requirements in less critical situations. Thus, if we required a similar accuracy in the bottom plot of Fig. 1, we would need to set the ETT correspondingly resulting in significantly more triggered events in less critical situations.

B. A robustness-proportional ETT regulation mechanism for inequality properties

In order to mitigate the disadvantages of the constant ETT policy, we propose a runtime ETT regulation mechanism based

on the robustness of inequality PL properties that can take advantage of less critical situations.

Definition 3. A static robustness-proportional ETT regulation mechanism for a signal $y_i \in \mathbf{y}_{\sim\varphi}$ for the inequality property φ^p of one of the types: $p(\hat{\mathbf{x}}_{\varphi}) > c$, $(p(\hat{\mathbf{x}}_{\varphi}) < c)$ is defined as:

$$\delta_{y_i, \varphi, t_{k+1}}(\hat{\mathbf{x}}_{t_k}, \epsilon_{y_i, \varphi}) = \frac{\rho(\varphi, \hat{\mathbf{x}}_{t_k})}{\epsilon_{y_i, \varphi}} \quad (19)$$

where $\epsilon_{y_i, \varphi} \in \mathbb{R}^+$.

We can now adapt the optimization problem in Eq. 17 such that Ξ becomes the set of defined $\epsilon_{y_i, \varphi^p}$ parameters and we add the constraint that $\epsilon_{y_i, \varphi^p} \in \mathbb{R}^+$.

Proposition 1. Using the ETT regulation mechanism in Definition 3, we achieve the property

$$\lim_{\rho(\varphi, \hat{\mathbf{x}}_{t_k}) \rightarrow 0} \delta_{y_i, \varphi, t_{k+1}}(\hat{\mathbf{x}}_{t_k}, \epsilon_{y_i, \varphi}) = 0 \quad (20)$$

which corresponds to smaller ETTs and thus improved accuracy in more critical situations.

For the remainder of this Section, we will focus on the ETT regulation mechanism in Definition 3 and the extension of this to arbitrary PL properties. We refer to this policy as ρ ETT. Later, in Section IV, we will revisit the TT approach in Eq. (18) and the CETT policy and compare these approaches to the ρ ETT policy presented in this Section. Next, we sketch the overall potential optimization approaches for determining suitable parameters for the TT, CETT and ρ ETT policies.

C. Parameter selection procedure outline

Determining the $\epsilon_{y_i, \varphi^p}$ parameters or constant ETTs in Eq. 17, can be categorized into two overall approaches; empirical and formal. The former approach entails running a large number of simulations for different parameter sets with random measurement noise and starting conditions. For example, if we consider the optimization problem for static ETTs in Eq. (17), we could start with all ETTs set to 0, and then gradually increase the ETTs until the property is violated. Conversely, if we consider the optimization problem in Eq (17) for ρ ETT, we would start with large $\epsilon_{y_i, \varphi^p}$ parameters and gradually decrease them until the property is violated. This may be a suitable approach if the system is complex and hard to formally analyze, but may have significant computational requirements. Alternatively, a suitable ETT can be determined by formally analyzing the system and controller dynamics. In this paper, we primarily consider the former empirical approach. In the next section, we show how to obtain a lower bound on the parameters such that when the uncertainty of the measured signals is taken into account then the ETTs will be reduced arbitrarily close to 0 as the system becomes unsafe. Later, in Section IV we provide insights on how parameter tuning of both constant ETTs and the proposed ETT regulation method can be performed. We also explore what effects suboptimal parameter configurations can have on the satisfaction of the property and the number of triggered events.

D. Guarantees for inequality property satisfaction detection

Recall that the robustness used to calculate the ETT in Definition 3 is an estimated robustness rather than the actual true robustness. As a result, the property in Proposition (1) does not guarantee that the ETTs approach zero when the robustness calculated on the true system state, becomes zero. Because, while the robustness may attain a positive value that is close to zero, because of its associated uncertainty, the real robustness may already have become negative. According to Eq. (14) and as Example 2 shows, we allow measurable signals to reside in known intervals. Similarly, the state estimates will also reside in an interval and consequently, the robustness will also reside in an interval which can be calculated using interval arithmetic. This robustness interval is then guaranteed to contain the true robustness. If the robustness interval has a negative lower bound, then we can conclude that the system is potentially unsafe allowing the controller (and the ETT regulation mechanism) to react appropriately.

Example 2 (Robustness interval). Consider the states $\hat{\mathbf{x}} = \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix}$ and the property $\varphi^p = (2x_1 + 4x_2) > 9$ and the corresponding robustness $\rho(\varphi^p, \hat{\mathbf{x}}_{t_k}) = 2x_1 + 4x_2 - 9$. At time t_k , we have the following estimated values of the signals $\hat{x}_{1, t_k} = 3, \hat{x}_{2, t_k} = 1$ corresponding to a robustness of $\rho(\varphi) = 2 \cdot 3 + 4 \cdot 1 - 9 = 1$. We can measure both x_1 and x_2 and use the ETT regulation mechanism from Definition 3 with the parameters $\epsilon_{x_1} = 3, \epsilon_{x_2} = 3$ providing us with ETTs $\delta_{x_1} = \frac{1}{3}, \delta_{x_2} = \frac{1}{3}$. We assume that the signals x_1 and x_2 do not change over the next time step and we apply the calculated ETT. Thus the robustness interval at time t_{k+1} becomes $2[3 - \frac{1}{3}, 3 + \frac{1}{3}] + 4[1 - \frac{1}{3}, 1 + \frac{1}{3}] = [10 - \frac{6}{3}, 10 + \frac{6}{3}] = [8, 12]$. As a result, the system could potentially violate φ without the remote state estimator or controller being aware which is undesirable.

In the following section, we provide a method for determining the set of minimum $\epsilon_{y_i, \varphi^p}$ parameters that ensure that the minimum ETT is always applied before the robustness calculated based on the true system state $\rho(\varphi^p, \mathbf{x}_{t_k})$ reaches zero.

Let $\rho(\varphi^p, \Delta\hat{\mathbf{x}}_{t_k})$ denote the robustness interval, where $\Delta\hat{\mathbf{x}}_{t_k}$ is the state estimate interval vector at time t_k . We assume that $\Delta\hat{\mathbf{x}}_{t_k}$ only contains directly measurable states and that the innovation update error (eq. (2)) is used. Thus $\Delta\hat{\mathbf{x}}_{t_k}$ can be constructed as follows:

$$\Delta\hat{\mathbf{x}}_{t_k} = \begin{bmatrix} [\hat{x}_{1, t_k} - \delta_{x_1, \varphi^p, t_k}(\cdot), \hat{x}_{1, t_k} + \delta_{x_1, \varphi^p, t_k}(\cdot)] \\ \vdots \\ [\hat{x}_{n, t_k} - \delta_{x_n, \varphi^p, t_k}(\cdot), \hat{x}_{n, t_k} + \delta_{x_n, \varphi^p, t_k}(\cdot)] \end{bmatrix}. \quad (21)$$

We also assume that it is possible to predict the bounds of the state estimate and consequently the robustness at the next time step $\Delta\hat{\mathbf{x}}_{t_{k+1}|t_k}$, given the state estimate information at time step t_k :

$$[\rho(\varphi^p, \Delta\hat{\mathbf{x}}_{t_{k+1}|t_k}), \bar{\rho}(\varphi^p, \Delta\hat{\mathbf{x}}_{t_{k+1}|t_k})] = \rho(\varphi^p, \Delta\hat{\mathbf{x}}_{t_{k+1}|t_k}). \quad (22)$$

This prediction can be obtained using set-membership filtering [28]. Alternatively, one may use a probabilistic state estimation approach such as the KF and use the associated uncertainty related to the state estimate, to construct a confidence interval on the state interval. In Theorem 1, we present a method for automatically determining the $\epsilon_{y_i, \varphi^p}$ parameters from Definition 3 such that the sign of the robustness will always equal the sign of the true robustness, i.e. we guarantee that we detect the satisfaction/violation of a property.

Theorem 1. Consider an inequality property φ^p with the robustness interval width $w(\rho(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}}))$ and the robustness of φ^p calculated on the true system state $\rho(\varphi^p, \mathbf{x}_{t_{k+1}})$ at time t_{k+1} . By setting $\epsilon_{y_i, \varphi^p} = \frac{\partial w(\rho(\varphi^p, \Delta \hat{\mathbf{x}}))}{\partial \delta_{y_i, \varphi^p, t_{k+1}}(\cdot)} \lambda_{y_i, \varphi^p} \epsilon_{\rho, \varphi^p}$ where $\sum_{y_i \in \mathbf{y}_{\sim \varphi^p}} \frac{1}{\lambda_{y_i, \varphi^p}} = 1, \epsilon_{\rho, \varphi^p} \geq 1$ and using $\underline{\rho}(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k)$ instead of $\rho(\varphi^p, \hat{\mathbf{x}}_{t_k})$ in Definition 3, we achieve the property

$$(\rho(\varphi^p, \hat{\mathbf{x}}_{t_k}) > 0) \Leftrightarrow (\rho(\varphi^p, \mathbf{x}_{t_k}) > 0) \quad (23)$$

when $\forall x_i \in \hat{\mathbf{x}} \quad \frac{\partial p(\hat{\mathbf{x}})}{\partial x_i} = \alpha_i$, the innovation update error is used and $\forall x_i \in \hat{\mathbf{x}} : \frac{\partial p(\hat{\mathbf{x}})}{\partial x_i} \neq 0 \exists y_i \in \mathbf{y}_{\sim \varphi^p} : y_i \equiv x_i$. The latter formalizes that all relevant states are directly measurable.

Proof. To satisfy the condition in eq. (23) we predict the robustness interval at time t_{k+1} to determine the necessary ETTs and thus signal intervals at time t_{k+1} . As the true robustness $\rho(\varphi^p, \mathbf{x}_{t_{k+1}}) \in \rho(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k)$, we consider the worst-case scenario ($\rho(\varphi^p, \mathbf{x}_{t_{k+1}}) = \underline{\rho}(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k)$). Next, we demonstrate that by setting $\epsilon_{y_i, \varphi^p} = \frac{\partial w(\rho(\varphi^p, \Delta \hat{\mathbf{x}}))}{\partial \delta_{y_i, \varphi^p, t_{k+1}}(\cdot)} \lambda_{y_i, \varphi^p} \epsilon_{\rho, \varphi^p}$ we have:

$$w(\rho(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}})) \leq \max\left(\frac{\underline{\rho}(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k)}{\epsilon_{\rho, \varphi^p}}, 0\right). \quad (24)$$

As the robustness is a linear combination of the bounded scaled state estimates and we use the innovation update error, the upper bound of the robustness interval width (the case when no events are triggered) can be calculated using the state estimate intervals $\Delta \hat{\mathbf{x}}_{t_{k+1}}$ from eq. (21). Using the result of Lemma 2, we have $w(\rho(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}})) = 2|\alpha_1| \delta_{y_1, \varphi^p, t_{k+1}}(\cdot) + \dots + 2|\alpha_n| \delta_{y_n, \varphi^p, t_{k+1}}(\cdot)$ and thus also $\frac{\partial w(\rho(\varphi^p, \Delta \hat{\mathbf{x}}))}{\partial \delta_{y_i, \varphi^p, t_{k+1}}(\cdot)} = 2|\alpha_i|$ where $\alpha_i = \frac{\partial \rho(\varphi^p, \hat{\mathbf{x}}_{t_{k+1}})}{\partial y_i}$. If we insert $\epsilon_{y_i, \varphi^p} = \frac{\partial w(\rho(\varphi^p, \Delta \hat{\mathbf{x}}))}{\partial \delta_{y_i, \varphi^p, t_{k+1}}(\cdot)} \lambda_{y_i, \varphi^p} \epsilon_{\rho, \varphi^p}$ in Definition 3 and use $\underline{\rho}(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k)$ instead of $\rho(\varphi^p, \hat{\mathbf{x}}_{t_k})$ we have:

$$\begin{aligned} & w(\rho(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}})) \\ &= \sum_{y_i \in \mathbf{y}_{\sim \varphi^p}} 2|\alpha_i| \frac{\max(\underline{\rho}(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k), 0)}{2|\alpha_i| \lambda_{y_i, \varphi^p} \epsilon_{\rho, \varphi^p}}, \\ &= \frac{1}{\epsilon_{\rho, \varphi^p}} \sum_{y_i \in \mathbf{y}_{\sim \varphi^p}} \frac{\max(\underline{\rho}(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k), 0)}{\lambda_{y_i, \varphi^p}} \\ &= \frac{\max(\underline{\rho}(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k), 0)}{\epsilon_{\rho, \varphi^p}}. \end{aligned}$$

Next, assume $\rho(\varphi^p, \hat{\mathbf{x}}_{t_{k+1}}) \leq 0$. Since $\rho(\varphi^p, \hat{\mathbf{x}}_{t_{k+1}}) \in \rho(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}})$ and $\rho(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}}) \subseteq \rho(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k)$, and

therefore $\underline{\rho}(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k) \leq \rho(\varphi^p, \hat{\mathbf{x}}_{t_{k+1}}) \leq 0$. Consequently

$$(w(\rho(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}})) = 0) \Rightarrow (\rho(\varphi^p, \hat{\mathbf{x}}_{t_{k+1}}) = \rho(\varphi^p, \mathbf{x}_{t_{k+1}}))$$

and thus $(\rho(\varphi^p, \hat{\mathbf{x}}_{t_{k+1}}) \leq 0) \Rightarrow (\rho(\varphi^p, \mathbf{x}_{t_{k+1}}) \leq 0)$. Following the same line of arguments, we can obtain $(\rho(\varphi^p, \mathbf{x}_{t_{k+1}}) \leq 0) \Rightarrow (\rho(\varphi^p, \hat{\mathbf{x}}_{t_{k+1}}) \leq 0)$. If $(\rho(\varphi^p, \hat{\mathbf{x}}_{t_k}) \leq 0) \Leftrightarrow (\rho(\varphi^p, \mathbf{x}_{t_k}) \leq 0)$, then $(\rho(\varphi^p, \hat{\mathbf{x}}_{t_k}) > 0) \Leftrightarrow ((\rho(\varphi^p, \mathbf{x}_{t_k})) > 0)$ which concludes the proof. \square

Do note that the ETT regulation method in Theorem 1 does not solve the optimization problem in Eq. (17), but rather it provides a method to determine a sufficiently small ETT such that the system is always aware of whether or not it is potentially violating a property and can act accordingly. The $\epsilon_{\rho, \varphi^p}$ parameter in Theorem 1, determines how early the robustness interval starts converging. Larger values of $\epsilon_{\rho, \varphi^p}$ correspond to generally lower ETTs and vice versa. We can then adapt the optimization problem in 17 by setting Ξ to the set of defined λ_{y_i, φ^p} and $\epsilon_{\rho, \varphi^p}$ parameters and adding the constraints in Theorem 1. We provide a default way to set the λ_{y_i, φ^p} weights from Theorem 1, in Proposition 2.

Proposition 2. Consider the λ_{y_i, φ^p} parameters in Theorem 1 where we require $\sum_{y_i \in \mathbf{y}_{\sim \hat{\mathbf{x}}}} \frac{1}{\lambda_{y_i, \varphi^p}} = 1$, and $\dim(\{x_i \in \hat{\mathbf{x}} : \frac{\partial p(\hat{\mathbf{x}})}{\partial x_i} \neq 0\}) = \dim(\mathbf{y}_{\sim \varphi^p})$ (a consequence of the relevant states being directly measurable) where $\dim(\mathbf{F})$ is the number of elements in the set \mathbf{F} . If we set $\lambda_{y_i, \varphi^p} = \dim(\mathbf{y}_{\sim \varphi^p}) \forall y_i \in \mathbf{y}_{\sim \varphi^p}$ then $\sum_{y_i \in \mathbf{y}_{\sim \varphi^p}} \frac{1}{\dim(\mathbf{y}_{\sim \varphi^p})} = 1$.

Next, we briefly discuss a few remarks regarding some of the assumptions related to Theorem 1 followed by how to handle the case when multiple inequality properties depend on one or more of the same signals.

1) *SOD update error and non-linearities:* Instead of the innovation-based update error, the SOD update error could also be used to bound the state used to calculate the robustness by using the interval determined by $\Delta \hat{x}_{i, t_k} = [y_{i, \tau_m^i} - \delta_{y_i, \varphi^p, t_k}, y_{i, \tau_m^i} + \delta_{y_i, \varphi^p, t_k}]$ where y_{i, τ_m^i} is the value of y_i at time the previous event-triggering time τ_m^i for the i^{th} sensor.

Additionally, if the function $p(\hat{\mathbf{x}})$ contains non-linear combinations of any elements of $\hat{\mathbf{x}}$, the guarantees that Theorem 1 provide become more difficult to achieve as we need to consider both the values that can be measured (bounded by Eq. (22)), as well as the state estimate intervals in the case that events are not triggered. However, we still note that it is likely beneficial to consider the partial derivative of the ETT w.r.t to the robustness interval width when choosing the $\epsilon_{y_i, \varphi^p}$ parameters in such cases as this gives a measure of how ‘‘important’’ a given signal is.

2) *Multiple properties with overlapping signal sets:* We now consider the case where multiple inequality PL properties are present which depend on overlapping signal sets.

We introduce the notation $\delta_{y_i, t_k}(\hat{\mathbf{x}}, \Phi)$ to denote the final applied ETT to the signal y_i at time t_k based on all system properties Φ . We rely on the assumption that if

$$\begin{aligned} & \forall y_i \in \mathbf{y}_{\sim \varphi_b}, \forall t_k \in [t_0, t_{end}] \delta_{y_i, t_k}(\hat{\mathbf{x}}, \Phi) = \delta_{y_i, \varphi_b, t_k}(\cdot) \\ & \rightarrow \forall t_k \in [t_0, t_{end}] \rho(\varphi_b, \hat{\mathbf{x}}_{t_k}) > 0, \end{aligned}$$

then

$$\begin{aligned} \forall y_i \in \mathbf{y}_{\sim\varphi_b}, \forall t_k \in [t_0, t_{end}], \\ \forall \delta_{y_i, t_k}(\hat{\mathbf{x}}, \Phi) \in \{\delta_{y_i, \varphi_a, t_k}(\cdot) : \delta_{y_i, \varphi_a, t_k}(\cdot) \leq \delta_{y_i, \varphi_b, t_k}(\cdot)\} \\ \rightarrow \forall t_k \in [t_0, t_{end}] \rho(\varphi_b, \hat{\mathbf{x}}_{t_k}) > 0, \end{aligned} \quad (25)$$

meaning that an $\delta_{y_i, \varphi_a, t_k}(\cdot)$ for a signal y_i defined based on the property φ_a , resulting in a smaller ETT than $\delta_{y_i, \varphi_b, t_k}(\cdot)$ defined based on another property φ_b , when applied to signal y_i , still guarantees the satisfaction of the other property φ_b .

Definition 4. For a set of PL properties Φ with either constant ETTs or ETTs calculated using Definition 3 with overlapping state estimate dependencies, the final ETT of a signal $y_i \in \mathbf{y}$ is given by:

$$\delta_{y_i, t_k}(\hat{\mathbf{x}}, \Phi) = \min_{\varphi_j \in \Phi} \delta_{y_i, \varphi_j, t_k}(\cdot) \text{ s.t. } y_i \in \mathbf{y}_{\sim\varphi_j}. \quad (26)$$

Eq. (26) is to be applied after any of the previous or following ETT regulation mechanisms have been applied if multiple separate properties are monitored.

Lemma 3. Definition 4 preserves Theorem 1, if Theorem 1 is used to regulate the ETT of the underlying inequality properties and the inequality properties satisfy the conditions in Theorem 1.

Proof. Using the result of Lemma 2, we know that $\frac{\partial w(\rho(\varphi, \Delta\hat{\mathbf{x}}))}{\partial \delta_{y_i, \varphi, t_k}(\cdot)} = 2|\alpha_i|$, meaning that if we decrease the ETT, then $w(\rho(\varphi, \Delta\hat{\mathbf{x}}_{t_k}))$ decreases as well which preserves the inequality. \square

E. ETT regulation for PL properties with inequality sub-properties

We now leverage the satisfaction relation of the remaining PL operators to define an ETT regulation mechanism that refines the ETT determined by the underlying inequality property, based on the propositional operators. For the remainder of the paper, we use the shorthand notation ρETT to denote the application of the robustness-relative ETT regulation policy based on Definition 3 and $\underline{\rho} ETT$ to denote the ETT regulation policy based on Theorem 1. In this Section, we consider the ρETT policy, and later in Section III-G we extend the $\underline{\rho} ETT$ from Theorem 1 to arbitrary PL properties. For simplicity and to facilitate intuition, the sub-properties of all propositional properties are initially restricted to inequality properties. Later in this section, we provide a way to compute the ETTs for an arbitrary PL property.

1) *The \wedge operator:* By Definition 2, for a property $\varphi_\wedge = \varphi_1 \wedge \varphi_2$ we desire to find an ETT policy ρETT that ensures

$$\begin{aligned} \rho ETT \Rightarrow (\rho(\varphi, \hat{\mathbf{x}}_{t_k}) > 0) \\ \equiv (\rho ETT \Rightarrow (\rho(\varphi_1, \hat{\mathbf{x}}_{t_k}))) \wedge \\ (\rho ETT \Rightarrow (\rho(\varphi_2, \hat{\mathbf{x}}_{t_k}))), \\ \forall t_k \in [t_0, t_{end}]. \end{aligned} \quad (27)$$

As a result, we can treat the analysis of the sub-properties φ_1 and φ_2 to determine suitable static ETTs or ETT regulation parameters, separately.

2) *The \vee operator:* According to Definition 3, for the property $\varphi_\vee = \varphi_1 \vee \varphi_2$, we desire to find an ETT policy ρETT such that

$$\begin{aligned} \rho ETT \Rightarrow (\rho(\hat{\mathbf{x}}_{\varphi_\vee, t_k}, \varphi_\vee) > 0) \\ \equiv \rho ETT \Rightarrow \\ [(\rho(\hat{\mathbf{x}}_{\varphi_1, t_k}, \varphi_1) > 0) \vee (\rho(\hat{\mathbf{x}}_{\varphi_2, t_k}, \varphi_2) > 0)], \\ \forall t_k \in [t_0, t_{end}]. \end{aligned} \quad (28)$$

This is a notable difference from the \wedge operator. We could treat the φ_1 and φ_2 sub-properties separately as we did with the \wedge operator since $(\varphi_1 \wedge \varphi_2) \Rightarrow (\varphi_1 \vee \varphi_2)$ but we then loose out on potential communication savings as only one property has to be satisfied at any given time. According to Eq. (28), the robustness of one sub-property can become negative, while the overall property is still satisfied. Thus we need to define how the ETT for a sub-property with negative robustness is calculated. We argue that when an inequality property has negative robustness, the ETT should be equal to 0 providing the system with the best possible accuracy of states related to that specific property. Thus we refine Definition 3 by adding:

$$\delta_{y_i, \varphi^p, t_{k+1}}^+(\cdot) = \max(\delta_{y_i, \varphi^p, t_{k+1}}(\cdot), 0). \quad (29)$$

We wish to leverage the relaxation in Eq. (28) to enlarge the ETTs of the inequality sub-property that requires the best accuracy (and thus the most measurements) to satisfy. We intuitively want to avoid directly mixing ETTs as different state estimates can have different ranges of values and thus the properties have different ranges of robustness and ETTs. Later in Example 2, we provide some intuition for this decision. To this end, we propose to use the normalized robustness ζ_ρ which is defined in Definition 5, to represent the ‘‘criticality’’ of an PL inequality property which is then later applied in Definition 6 to enlarge the ETT of the less critical inequality property.

Definition 5 (Normalized robustness ζ_ρ). We define the normalized robustness for an inequality property φ^p as:

$$\zeta_\rho(\varphi^p, \hat{\mathbf{x}}_{t_k}) = \frac{\max(\rho(\varphi^p, \hat{\mathbf{x}}_{t_k}), 0)}{\rho_{\max}(\varphi^p)},$$

where $\rho_{\max}(\varphi^p) > 0$ and is the maximum possible robustness of the inequality property φ^p .

We now define the ETT refinement mechanism for propositional PL properties in Definition 6.

Definition 6. We define the ETT regulation mechanism for propositional property φ with inequality sub-properties as a refinement $\bar{\delta}_{y_i, \varphi, t_{k+1}}(\hat{\mathbf{x}}_{t_k}, \beta_{\varphi, t_k})$ of $\delta_{y_i, \varphi, t_{k+1}}^+(\cdot)$ from eq. (29)

for a signal y_i :

$$\bar{\delta}_{y_i, \varphi, t_{k+1}}(\hat{\mathbf{x}}_{t_k}, \beta_{\varphi, t_k}) = \begin{cases} \min_{\varphi' \in \{\varphi_1^p, \varphi_2^p\}} (\delta_{y_i, \varphi', t_{k+1}}^{\in}(\hat{\mathbf{x}}_{t_k}, \epsilon_{y_i, \varphi'}, \beta_{\varphi, t_k})), \\ \quad \text{if } \varphi = \varphi_1^p \vee \varphi_2^p, \\ \min_{\varphi' \in \{\varphi_1^p, \varphi_2^p\}} (\delta_{y_i, \varphi', t_{k+1}}^{\in}(\hat{\mathbf{x}}_{t_k}, \epsilon_{y_i, \varphi'}, 0)) \\ \quad \text{if } \varphi = \varphi_1^p \wedge \varphi_2^p, \end{cases}$$

$$\text{where } \beta_{\varphi, t_k} = \max_{\varphi' \in \{\varphi_1^p, \varphi_2^p\}} (\zeta_{\rho}(\varphi', \hat{\mathbf{x}}_{t_k})),$$

$$\delta_{y_i, \varphi^p, t_{k+1}}^{\in}(\hat{\mathbf{x}}_{t_k}, \epsilon_{y_i, \varphi^p}, \beta_{\varphi, t_k}) = \begin{cases} \delta_{y_i, \varphi^p, t_{k+1}}^+(\cdot) \\ \quad + \max(\beta_{\varphi, t_k} - \zeta_{\rho}(\varphi^p, \hat{\mathbf{x}}_{t_k}), 0) \frac{\rho_{max}(\varphi^p)}{\epsilon_{y_i, \varphi^p}} \\ \quad \text{if } y_i \in \mathbf{y}_{\sim \varphi^p}, \\ \infty \quad \text{otherwise,} \end{cases}$$

Definition 4 is applied within Definition 6 using the min operator for each sub-property. If more than one property is present, Definition 4 should be applied externally.

Example 3 (Normalized robustness). Consider the property $\varphi_{\zeta_{\rho}} = (x_1 < 1) \vee (x_2 > 1000)$ where the value of $x_1 \in [0, 1.5]$ and the value of $x_2 \in [-1000, 3000]$. We use the parameters $\epsilon_{x_1, \varphi_{\zeta_{\rho}}} = \epsilon_{x_2, \varphi_{\zeta_{\rho}}} = 5$. At time t_k , $x_{1, t_k} = 1.2$ and $x_{2, t_k} = 1500$, resulting in robustnesses of $\rho(x_1 < 1) = 1 - 1.2 = -0.2$ and $\rho(x_2 > 1000) = 1500 - 1000 = 500$ and ETTs: $\delta_{x_1, \varphi_{\zeta_{\rho}}, t_{k+1}}(\cdot) = \frac{0}{5} = 0$ and $\delta_{x_2, \varphi_{\zeta_{\rho}}, t_{k+1}}(\cdot) = \frac{500}{5} = 100$. As $(x_2 > 1000)$ the overall property is satisfied, but we want to utilize this fact to relax $\delta_{x_1, \varphi_{\zeta_{\rho}}, t_{k+1}}(\cdot)$ as setting $\delta_{x_1, \varphi_{\zeta_{\rho}}, t_{k+1}}(\cdot) = 0$ will likely result in many triggered events. Directly setting $\delta_{x_1, \varphi_{\zeta_{\rho}}, t_{k+1}}(\cdot) = \delta_{x_2, \varphi_{\zeta_{\rho}}, t_{k+1}}(\cdot) = 100$ is likely too large and does not preserve the meaningful range of ETTs for $\delta_{x_1, \varphi_{\zeta_{\rho}}, t_{k+1}}(\cdot)$. Using Definitions 5 and 6, we instead obtain: $\zeta_{\rho}(x_1 < 1, -1.2) = 0$, $\zeta_{\rho}(x_2 > 1000, 1500) = \frac{500}{2000}$. Refining $\delta_{x_1, \varphi_{\zeta_{\rho}}, t_{k+1}}(\cdot)$ gives us: $\bar{\delta}_{x_1, \varphi_{\zeta_{\rho}}, t_{k+1}}(\cdot) = \delta_{x_1, \varphi_{\zeta_{\rho}}, t_{k+1}}(\cdot) + (\frac{500}{2000} - 0) \frac{1}{5} = 0 + \frac{1}{4} \frac{1}{5} = 0.05$ and $\bar{\delta}_{x_2, \varphi_{\zeta_{\rho}}, t_{k+1}}(\cdot) = \delta_{x_2, \varphi_{\zeta_{\rho}}, t_{k+1}}(\cdot)$. Thus Definition 5 gives us a way to translate/relate the ‘‘criticality’’ of properties between each other.

In the following Section, we adapt Definition 6 to arbitrary PL properties.

F. ETT regulation for arbitrary PL properties

1) *Negation of PL properties*: To regulate the ETT using Definition 3 for an arbitrary PL property which potentially includes negations of logical operators, we transform the property into its Negation Normal Form (NNF), where the negations are propagated to the underlying inequality properties. The robustness of the negated inequality is then used to regulate the ETT according to Definition 3.

Example 4 (Negation Normal Form). Consider the PL property $\varphi_{NNF} = (\varphi_1 \wedge \neg(\varphi_2 \rightarrow \varphi_3))$ where φ_1, φ_2 and φ_3 are

all propositional PL properties. We propagate the negation as follows

$$\begin{aligned} & \neg(\varphi_2 \rightarrow \varphi_3) \\ & \equiv \neg\neg\varphi_2 \vee \neg\varphi_3 \\ & \equiv \varphi_2 \vee \neg\varphi_3, \end{aligned}$$

which gives us the final property $\varphi_{NNF} \equiv \varphi_1 \wedge (\varphi_2 \vee \neg\varphi_3)$.

Initially, we need to define how to calculate $\zeta_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k})$ for an arbitrary PL property.

Definition 7. The Normalized robustness $\tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k})$ for an arbitrary PL property φ is defined recursively as:

$$\tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k}) = \begin{cases} \max(\tilde{\zeta}_{\rho}(\varphi_1, \hat{\mathbf{x}}_{t_k}), \tilde{\zeta}_{\rho}(\varphi_2, \hat{\mathbf{x}}_{t_k})) & \text{if } \varphi = (\varphi_1 \vee \varphi_2), \\ \min(\tilde{\zeta}_{\rho}(\varphi_1, \hat{\mathbf{x}}_{t_k}), \tilde{\zeta}_{\rho}(\varphi_2, \hat{\mathbf{x}}_{t_k})) & \text{if } \varphi = (\varphi_1 \wedge \varphi_2), \\ \zeta_{\rho}(\varphi^p, \hat{\mathbf{x}}_{t_k}) & \text{otherwise,} \end{cases}$$

where $\zeta_{\rho}(\varphi^p, \hat{\mathbf{x}}_{t_k})$ is calculated according to Definition 5.

Next, we present a property of Definition 7 that will be useful in proving Theorem 1 for propositional and later arbitrary PL properties.

Lemma 4. $(\tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k}) > 0) \Leftrightarrow (\rho(\varphi, \hat{\mathbf{x}}_{t_k}) > 0)$.

Proof. We prove each case in Definition 7, below.

otherwise ($\varphi = \varphi^p$): It follows from Definition 5 that $\tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k})$ preserves the sign since $\rho_{max} > 0$.

$\varphi = \varphi_1 \vee \varphi_2$: Assume $\tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k}) > 0$. Then by Definition 7 ($\tilde{\zeta}_{\rho}(\varphi_1, \hat{\mathbf{x}}_{t_k}) > 0$) \vee ($\tilde{\zeta}_{\rho}(\varphi_2, \hat{\mathbf{x}}_{t_k}) > 0$). Consequently, we have $(\rho(\varphi_1, \hat{\mathbf{x}}_{t_k}) > 0) \vee (\rho(\varphi_2, \hat{\mathbf{x}}_{t_k}) > 0)$, which through Definition 2 results in $\rho(\varphi, \hat{\mathbf{x}}_{t_k})$. Thus $(\tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k}) > 0) \Rightarrow (\rho(\varphi, \hat{\mathbf{x}}_{t_k}) > 0)$. Following the reverse procedure, we can obtain $\rho(\varphi, \hat{\mathbf{x}}_{t_k}) \Rightarrow \tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k})$.

$\varphi = \varphi_1 \wedge \varphi_2$: The proof for this case can be obtained following the same procedure as for the \vee operator.

Thus for any arbitrary propositional property φ , $(\tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k}) > 0) \Leftrightarrow (\rho(\varphi, \hat{\mathbf{x}}_{t_k}) > 0)$. \square

After determining $\tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k})$ we need a slight adaptation of Definition 6 to account for nested propositional operators which we provide in Definition 8.

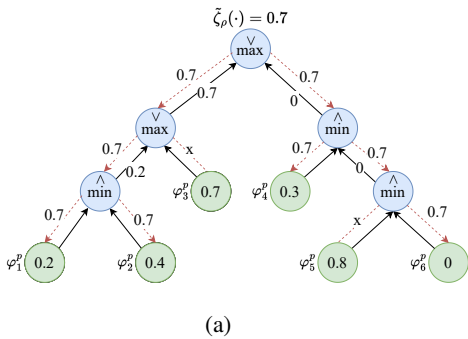
Definition 8 (Arbitrary propositional properties ETT refinement). Consider the arbitrary propositional property φ . We define the ETT refinement $\bar{\delta}_{y_i, \varphi, t_{k+1}}(\hat{\mathbf{x}}_{t_k}, \beta_{\varphi, t_k})$ for a signal

$y_i \in \mathbf{y} \sim \varphi$ at time t_{k+1} :

$$\tilde{\delta}_{y_i, \varphi, t_{k+1}}^{\epsilon}(\hat{\mathbf{x}}_{t_k}, \beta_{\varphi, t_k}) = \begin{cases} \delta_{y_i, \varphi, t_{k+1}}^{\epsilon}(\hat{\mathbf{x}}_{t_k}, \epsilon_{y_i, \varphi}, \beta_{\varphi, t_k}) & \text{if } \varphi = \varphi^p, \\ \min_{\varphi' \in \{\varphi_1, \varphi_2\}} (\tilde{\delta}_{y_i, \varphi', t_{k+1}}^{\epsilon}(\hat{\mathbf{x}}_{t_k}, \beta_{\varphi, t_k})) & \text{if } \varphi = \varphi_1 \wedge \varphi_2, \\ \min_{(\varphi', \varphi'') \in \{(\varphi_1, \varphi_2), (\varphi_2, \varphi_1)\}} (\tilde{\delta}_{y_i, \varphi', t_{k+1}}^{\epsilon}(\hat{\mathbf{x}}_{t_k}, \max(\beta_{\varphi, t_k}, \tilde{\zeta}_{\rho}(\varphi'', \hat{\mathbf{x}}_{t_k}))) & \text{if } \varphi = \varphi_1 \vee \varphi_2, \end{cases}$$

where $\delta_{y_i, \varphi, t_{k+1}}^{\epsilon}(\hat{\mathbf{x}}_{t_k}, \epsilon_{y_i, \varphi}, \beta_{\varphi, t_k})$ is defined in Definition 6 and we initialize $\beta_{\varphi, t_k} = \tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k})$ of the top-level property φ . The min operations take care of potentially multiple defined ETTs for a given signal, and thus Definition 4 is implicitly applied.

An example of the application of Definition 7 and the propagation of $\tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k})$ in Definition 8 is visualized in Fig. 3.



(a)

Fig. 3: A binary tree representation of the PL property $((\varphi_1 \wedge \varphi_2) \vee \varphi_3) \vee (\varphi_4 \wedge (\varphi_5 \wedge \varphi_6))$. The solid black arrows indicate the application of Definition 7 and the red dotted arrows indicate the applied β value in Definition 8 for each inequality property.

Definition 8 concludes the proposed ETT regulation mechanism ρETT . In the following Section, we extend the property in Theorem 1 arbitrary PL properties using a modified version of ρETT .

G. Providing detection guarantees for arbitrary PL properties

We start by adapting several Definitions from Sections III-E and III-F. We denote the ETT ≥ 0 calculated using the result of Theorem 1 as:

$$\underline{\delta}_{y_i, \varphi^p, t_{k+1}}^+ (\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k) = \frac{\max(\underline{\rho}(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k), 0)}{\frac{\partial w(\rho(\varphi^p, \Delta \hat{\mathbf{x}}))}{\partial \delta_{y_i, \varphi^p, t_{k+1}}^+} \lambda_{y_i, \varphi^p} \epsilon_{\rho, \varphi^p}}. \quad (30)$$

Next, we adapt the normalized robustness from Definitions 5 and 7.

Definition 9. We define the worst-case normalized robustness of an inequality property φ^p as:

$$\underline{\zeta}_{\rho}(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k) = \frac{\max(\underline{\rho}(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k), 0)}{\rho_{max}(\varphi^p)}$$

Definition 10. We define the worst-case normalized robustness $\tilde{\zeta}_{\rho}(\varphi, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k)$ of an arbitrary PL property as and adaptation of $\tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k})$ as follows:

$$\tilde{\zeta}_{\rho}(\varphi, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k) = \begin{cases} \max_{\varphi' \in \{\varphi_1, \varphi_2\}} \tilde{\zeta}_{\rho}(\varphi', \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k) & \text{if } \varphi = \varphi_1 \vee \varphi_2, \\ \min_{\varphi' \in \{\varphi_1, \varphi_2\}} \tilde{\zeta}_{\rho}(\varphi', \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k) & \text{if } \varphi = \varphi_1 \wedge \varphi_2, \\ \underline{\zeta}_{\rho}(\varphi, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k) & \text{if } \varphi = \varphi^p. \end{cases}$$

Next, we adapt $\delta_{y_i, \varphi^p, t_{k+1}}^{\epsilon}(\hat{\mathbf{x}}_{t_k}, \epsilon_{y_i, \varphi^p}, \beta_{\varphi, t_k})$ from Definition 6.

Definition 11. We define an adapted version of $\delta_{y_i, \varphi^p, t_{k+1}}^{\epsilon}(\hat{\mathbf{x}}_{t_k}, \epsilon_{y_i, \varphi^p}, \beta_{\varphi, t_k})$ from Definition 6 to help provide the guarantees in Theorem 1.

$$\delta_{y_i, \varphi, t_{k+1}}^{\epsilon}(\Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k, \epsilon_{y_i, \varphi}, \beta_{\varphi, t_{k+1}} | t_k) = \begin{cases} \delta_{y_i, \varphi^p, t_{k+1}}^+ (\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k) + \max(\beta_{\varphi, t_{k+1}} | t_k - \underline{\zeta}_{\rho}(\varphi^p, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k), 0) \frac{\rho_{max}(\varphi^p)}{\epsilon_{y_i, \varphi^p}} & \text{if } y_i \in \mathbf{y} \sim \varphi^p, \\ \infty & \text{otherwise.} \end{cases}$$

We are now ready to present ρETT for arbitrary PL properties. Due to space constraints, we do not provide the explicit definition, but instead list the changes necessary to Definition 8, in Theorem 2.

Theorem 2. Definition 8 ensures

$$(\rho(\varphi, \hat{\mathbf{x}}_{t_k}) > 0) \Leftrightarrow (\rho(\varphi, \mathbf{x}_{t_k}) > 0) \quad (31)$$

for an arbitrary propositional property φ if $\tilde{\zeta}_{\rho}(\varphi, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k)$ is used instead of $\tilde{\zeta}_{\rho}(\varphi, \hat{\mathbf{x}}_{t_k})$, the inequality properties satisfy the constraints in Theorem 1 and $\delta_{y_i, \varphi^p, t_{k+1}}^{\epsilon}(\Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k, \epsilon_{y_i, \varphi^p}, \beta_{\varphi, t_{k+1}} | t_k)$ is used instead of $\delta_{y_i, \varphi, t_{k+1}}^{\epsilon}(\hat{\mathbf{x}}_{t_k}, \epsilon_{y_i, \varphi}, \beta_{\varphi, t_k})$.

Proof. We prove that using the ETT regulation described in Theorem 2 ensures $(\rho(\varphi, \hat{\mathbf{x}}_{t_k}) \leq 0) \Leftrightarrow (\rho(\varphi, \mathbf{x}_{t_k}) \leq 0)$ which then guarantees $(\rho(\varphi, \hat{\mathbf{x}}_{t_k}) > 0) \Leftrightarrow (\rho(\varphi, \mathbf{x}_{t_k}) > 0)$. Using the same technique as in Lemma 4, we can show that

$$(\tilde{\zeta}_{\rho}(\varphi, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k) = 0) \Leftrightarrow (\underline{\rho}(\varphi, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k) \leq 0). \quad (32)$$

Now assume that $\rho(\varphi, \mathbf{x}_{t_{k+1}}) \leq 0$. Since $\rho(\varphi, \mathbf{x}_{t_{k+1}}) \in \rho(\varphi, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k)$, then $\underline{\rho}(\varphi, \Delta \hat{\mathbf{x}}_{t_{k+1}} | t_k) \leq 0$. We now show that the ETTs of the inequality property that produces the resulting robustness remain unaffected by the operations in Definition 6 in the case of negative robustness. We denote the overall property φ_o and the inequality property that produces the robustness value of the overall property, φ_o^p . We consider the \vee and \wedge operators in turn.

$\varphi = \varphi_1 \vee \varphi_2$: In order for φ_ρ^p to be a sub-property of φ , then $\rho(\varphi, \hat{\mathbf{x}}_{t_{k+1}}) = \rho(\varphi_\rho^p, \hat{\mathbf{x}}_{t_{k+1}})$. Since we assume $\rho(\varphi, \mathbf{x}_{t_{k+1}}) \leq 0$, this entails $\rho(\varphi_1, \hat{\mathbf{x}}_{t_{k+1}}) \leq 0$ and $\rho(\varphi_2, \hat{\mathbf{x}}_{t_{k+1}}) \leq 0$ which again entails $\tilde{\zeta}_\rho(\varphi, \Delta\hat{\mathbf{x}}_{t_{k+1}|t_k}) = 0$. Thus for the sub-property containing or equal to φ_ρ^p , the propagated $\underline{\beta}_{t_{k+1}|t_k}$ value is always equal to 0.

$\varphi = \varphi_1 \wedge \varphi_2$: According to Definition 6, $\underline{\beta}_{\varphi, t_{k+1}|t_k}$ is simply propagated and is equal to 0 if the sub-properties contain or are equal to φ_ρ^p .

Since the $\underline{\beta}_{t_k|t_{k-1}}$ value propagated to φ_ρ^p will always be 0 in the case of overall negative robustness the term $\max(\underline{\beta}_{t_{k+1}|t_k} - \tilde{\zeta}_\rho(\varphi_\rho^p, \Delta\hat{\mathbf{x}}_{t_{k+1}|t_k}), 0)$ in Definition 11 will equal 0 meaning that the ETTs related to φ_ρ^p are unchanged and thus satisfy Theorem 1 for the inequality property φ_ρ^p and thus for the overall property φ_o . \square

This concludes the presentation of our proposed ETT regulation mechanism for PL properties. In the following Section, we evaluate the TT, CETT, ρETT and ρETT in a simulated case study where we explore various parameter configurations and their impact on the robustness and number of triggered events.

IV. CASE STUDY

To evaluate our proposed threshold regulation mechanism, a simulated vehicle convoy in an adaptive cruise control (ACC) scenario is used. Initially, a single lane with two vehicles is considered to evaluate the ETT regulation mechanism in Definition 3 for a single inequality property. To evaluate more complex propositional properties, we extend the scenario to include an additional lane with faster vehicles. Fig. 4 shows an overview of the two setups. Next, we describe the simulation and scenario setup.

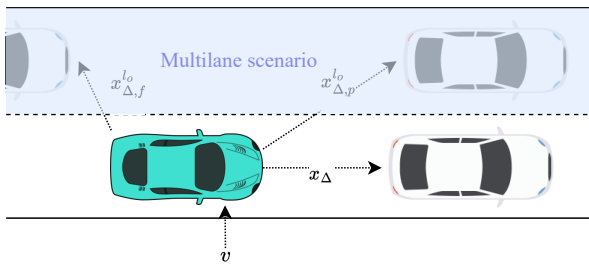


Fig. 4: An overview of the case-study simulated scenarios. The blue vehicle corresponds to the ACC vehicle which can measure its own speed (v), the distance to the preceding vehicle in the same lane x_p and the distance to the preceding $x_{l_o,p}$ and following $x_{l_o,f}$ vehicles in the fast lane in the multi-lane scenario.

A. Simulation setup

All vehicles in the convoy can be described by the linear discrete-time state-space model specified in Eq. 3.

The noise is assumed to be standard Gaussian noise with zero mean and the specified variance. For the remaining

vehicles, we use models without process noise or drift to control their evolution over time. The ACC vehicle tracks itself and all other relevant vehicles using the above model using a separate Kalman filter (KF) for each vehicle. The A and B matrices are the same for all instances and are given by

$$A = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}, B = \begin{bmatrix} \frac{1}{2}T_s^2 \\ T_s \end{bmatrix}, \quad (33)$$

where T_s is the sampling interval. As the ACC vehicle does not know the control input for the other vehicles, B can be disregarded when tracking other vehicles. The C matrices for the ACC vehicle and remaining tracked vehicles denoted with an o subscript, the process noise of the ACC vehicle w_{ACC} and other vehicles w_o , and the measurement noise of the speed r_v and distance r_{x_Δ} sensors are given by

$$C_{ACC} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C_o = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, w_{ACC} = \begin{bmatrix} 2.5 \cdot 10^{-9} & 5.0 \cdot 10^{-7} \\ 5.0 \cdot 10^{-7} & 1 \cdot 10^{-4} \end{bmatrix}, \\ w_o = 10 \cdot w_{ACC}, \quad r_{x_\Delta} = r_v = 0.1.$$

To control the ACC vehicle acceleration, we use the Intelligent Driver Model (IDM) [29]. The IDM attempts to keep a speed-dependent distance to the preceding vehicle given by $x_{ss}(d_0, v, T) = d_0 + vT$ where d_0 is a constant, v is the speed of the ACC vehicle and T is a parameter known as the time headway. We choose $d_0 = 2.7m$ and $T = 2s$.

The vehicles all start with the same initial speed of $30m/s$ and are placed at intervals of $x_{ss}(\cdot) + 20m$. The lead vehicle in the ACC vehicle lane has a predefined control schedule where it keeps a constant speed for the first 20 seconds as the ACC vehicle catches up whereafter it brakes with an acceleration of $a_{min} = -5m/s^2$ for 5 seconds and then accelerates to the original speed over 10 seconds at $a_{max} = 2.5m/s^2$ which corresponds to the minimum and maximum acceleration of the ACC vehicle respectively. Thus, the ACC vehicle will experience a variety of more and less safety-critical scenarios.

To determine the update error ($e(\cdot)$ in Eq. (1)), we use the innovation-based update error from Eq. 2. Since the prediction model of the preceding vehicle at the ACC vehicle lacks the control input information, the model is inherently more inaccurate and will lead to significantly more transmitted measurements compared to the sensor measuring the speed of the ACC vehicle. To add some importance to the speed sensor, we simulate an overestimation of the wind resistance leading to greater acceleration than expected.

For the sake of simplicity, we assume that the measurements, if they satisfy the event-triggering condition, are available instantly at the remote state estimator. Additionally, the information necessary to evaluate the event-triggering condition is assumed to always be available at the smart sensors without the need for additional communication. For single-signal properties, the latter assumption is valid, but it may not be the case for multiple-signal properties. We discuss this further in Section V.

In Section IV-B, we compare the TT, CETT, ρETT and ρETT approaches for an inequality property. Later in Section IV-C, we compare the TT, CETT and ρETT policies for a

more complex propositional property involving a multilane scenario.

B. Inequality Property for Single Lane Vehicle Convoy

As perfect knowledge of the preceding vehicle is not available, the controller will not be able to keep the desired distance $x_{ss}(\cdot)$ when the lead vehicle brakes. As a result, we construct our inequality property: $\varphi_a = x_\Delta > x_{ss}(d_\varphi, v, T)$ where $x_\Delta = x_p - x$, x_p and x are the position of the preceding and ACC vehicle, respectively, and $0 \leq d_\varphi < d_0$. The reason for choosing $d_\varphi < d_0$ is that the IDM will not be able to keep the desired distance to the lead vehicle as the ACC vehicle has imperfect knowledge of the speed and acceleration of the lead vehicle. Thus the target distance must be larger than the safe distance. We choose $d_\varphi = 0m$. This gives us the robustness:

$$\rho(\varphi, \hat{\mathbf{x}}_{t_k}) = x_{\Delta, t_k} - x_{ss}(0, v_{t_k}, T).$$

Initially, we consider the feasibility problem in Eq. (18), where we aim to find a sampling frequency that enables the system to satisfy the property. We test two sampling intervals T_s of 0.01s and 0.02 by running 20 simulations of the system for each configuration. We then determine the minimum overall robustness ρ_{min} calculated using the true system state produced for each configuration across simulations. Setting $T_s = 0.01s$ resulted in $\rho_{min} = \mathbf{0.36}$ while setting $T_s = 0.02s$ resulted in $\rho_{min} = \mathbf{-0.60}$ (negative robustness), wherefore we choose $T_s = 0.01s$ for all remaining experiments for all ETT regulation policies. Note that T_s could potentially be increased to somewhere between 0.01 and 0.02, but the number of transmissions cannot be reduced by more than a factor of two given the results of this test.

Next, we consider the CETT, ρETT , and $\underline{\rho} ETT$ policies for the inequality property φ_a . We choose $\mathbf{y}_{\sim\varphi_a} = \{v, x_\Delta\}$ and test a variety of different combinations of $\epsilon_{y_i, \varphi_a}$, λ_{y_i, φ_a} and $\epsilon_{\rho, \varphi_a}$ parameters and constant ETTs using a grid-search technique. We run 20 simulations for each set of parameters and the minimum robustness over all simulations is recorded in Table II. Table II shows that the ETT policies perform significantly better than the TT policy but also achieve a smaller ρ_{min} for the corresponding configuration. The ρETT policy triggers approximately **41.8%** fewer events compared to the CETT policy while the $\underline{\rho} ETT$ policy triggers **28.4%** fewer events than the CETT policy.

As the results in Table II are found using a brute-force parameter search, they are not guaranteed to be the global minimum. However, the difference in the number of triggered events between the TT, CETT and ρETT is large enough to demonstrate a significant advantage of the ρETT policy over the TT and CETT. Additionally, the ETT policies seem to lead to a smaller ρ_{min} value compared to the TT policy. In a real-world scenario, we may not be interested in the robustness being this close to zero, and we can instead revise the optimization problem in Eq. (17) to ensure that $\rho(\varphi, \hat{\mathbf{x}}_{t_k}) > \eta$ where $\eta > 0$.

Parameter exploration: For the ρETT regulation mechanism, we provide a contour plot depicting parameter combinations that result in a positive minimum robustness value

along with the resulting number of triggered events in Fig. 5. Fig. 5 shows that generally larger $\epsilon_{y_i, \varphi}$ values result in more triggered events regardless of the signal as we would expect. However, increasing the ϵ_{v, φ_a} value leads to a smaller increase in the number of triggered events compared to increasing $\epsilon_{x_\Delta, \varphi_a}$ by the same amount. The figure also shows that when either of the $\epsilon_{y_i, \varphi}$ parameters are below a certain threshold, larger values of the other $\epsilon_{y_i, \varphi}$ parameter will never result in positive robustness. Additionally, we see that a tradeoff is present in the lower left corner of the feasible set of solutions. Smaller values of either $\epsilon_{y_i, \varphi}$ parameter, require larger values for the other parameter to ensure satisfaction. Fig. 5 also shows that the assumption in Eq. (25) is always true for the $\epsilon_{x_\Delta, \varphi_a}$ signal but not necessarily for the ϵ_{v, φ_a} parameter. For certain parameter configurations, some larger values of the ϵ_{v, φ_a} (resulting in smaller ETTs) do not enable the controller to satisfy the property, while other smaller values of ϵ_{v, φ_a} do. This is likely due to the introduced random noise and the relatively small number of simulations. Due to space constraints, we do not provide the plot for the CETT policy, but the experiments conducted for the CETT policy show a similar tradeoff.

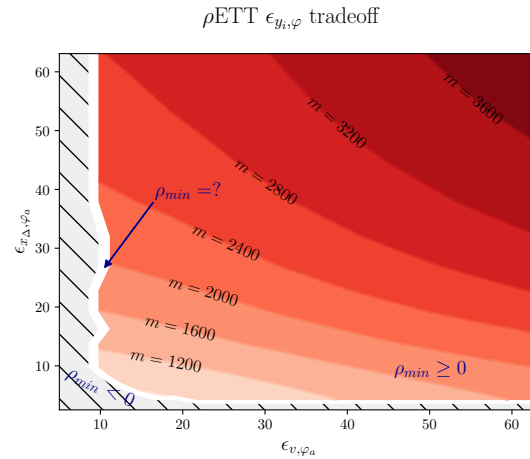


Fig. 5: A contour plot of ϵ_{v, φ_a} and $\epsilon_{x_\Delta, \varphi_a}$ parameter combinations where the red area indicates a feasible configuration and the grey hatched area indicates an infeasible configuration. The white area indicates untested parameter configurations. The shade of red indicates the average number of transmitted measurements.

In Fig. 6, we provide a plot that explores the tradeoff between the achieved minimum robustness and the number of transmitted measurements for all transmission and ETT regulation strategies. The results shown in Fig. 6 are the results corresponding to the Pareto front (i.e. the set of parameters that achieve a given ρ_{min} value with the fewest number of transmitted measurements). Fig. 6 shows that more transmitted measurements generally lead to larger ρ_{min} values for the Pareto optimal parameter configurations. However, the number of packets increases significantly as we achieve a larger ρ_{min} value resulting in a small performance gain per additional transmitted packet. Generally, the ρETT achieves the best tradeoff out of all ETT regulation mechanisms while the

ρ ETT policy produces slightly worse results in terms of the number of transmitted packets. This is to be expected since the ETT is regulated based on a more conservative state estimate. The percentage-wise reduction in the number of triggered events is somewhat consistent around **40%** for the ρ ETT policy compared to CETT which also applies to the best found configuration in Table II.

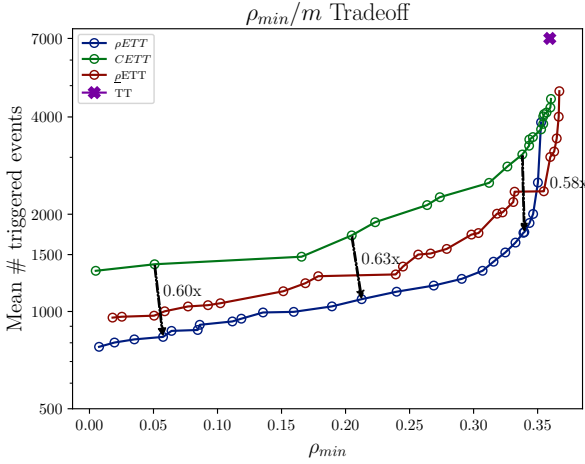


Fig. 6: A plot of non-dominated parameter configurations for the CETT and ρ ETT ETT regulation strategies, where m is the average number of transmitted packets and ρ_{min} is the minimum robustness evaluated on the true system state over all simulations for a given configuration. Several black arrows indicate what fraction of packets are needed for the ρ ETT method to achieve similar minimum robustness values compared to the CETT approach.

We now further investigate the ρ ETT policy. Fig. 7 depicts results for different values of $\epsilon_{\rho, \varphi}$ and $\lambda_{y_i, \varphi}$ combinations. Fig. 7 shows that even though the overall uncertainty is fixed relative to the robustness, different $\lambda_{y_i, \varphi}$ combinations produce both different average minimum robustness and a different number of triggered events. We see that for the most part, especially for lower $\epsilon_{\rho, \varphi}$ values, there is a large difference in the number of transmitted packets and some difference in resulting robustness for different $\lambda_{y_i, \varphi}$ parameter combinations. Furthermore, Fig. 7 shows the more transmitted packets do not necessarily lead to larger average ρ_{min} values if the parameters are configured sub-optimally. Despite the irregularities in Fig. 7, there could be some potential to apply a more intelligent numerical optimization algorithm to find locally optimal parameter configurations as there seems to be some structure to the parameter space and the cost in terms of ρ_{min} and the number of triggered events.

C. Propositional Property for Multi-lane Vehicle Convoy

To evaluate the ETT regulation mechanisms proposed in Section III-E, we extend the scenario from Section IV-B to a two-lane ACC scenario. We add a fast lane with additional faster vehicles and enable the ACC vehicle to measure the distance to the two closest vehicles ($x_{\Delta, p}^{l_o}, x_{\Delta, f}^{l_o}$ in preceding and following the ACC vehicle respectively) using two additional

Strategy	Parameters	φ_a (Single lane)	
		ρ_{min}	$m \pm \sigma(m)$
TT	$T_s = 0.01s$	0.36	7000
CETT	$\delta_{v, \varphi_a} = 0.16$ $\delta_{x_{\Delta, \varphi_a}} = 0.50$	0.005	1336 ± 27
ρ ETT	$\epsilon_{v, \varphi_a} = 16.64$ $\epsilon_{x_{\Delta}} = 4.95$	0.007	777 ± 14
ρ ETT	$\epsilon_{v, \varphi_a} = 13.38$ $\epsilon_{x_{\Delta}} = 3.95$	0.018	957 ± 14
		φ_b (Multilane critical)	
		ρ_{min}	$m \pm \sigma(m)$
TT	$T_s = 0.01s$	3.60	14000
CETT	$\delta_v = 0.16$ $\delta_{x_{\Delta}} = 0.50$ $\delta_{x_{\Delta, p}^{l_o}} = 0.50$ $\delta_{x_{\Delta, f}^{l_o}} = 2.00$	3.59	1527 ± 23
ρ ETT	$\epsilon_{v, \varphi_{b1}} = \epsilon_{v, \varphi_{b3}} = 16.64$ $\epsilon_{x_{\Delta, \varphi_{b1}}} = 4.95$ $\epsilon_{x_{\Delta, p}^{l_o}, \varphi_{b3}} = 4.95$ $\epsilon_{x_{\Delta, f}^{l_o}, \varphi_{b2}} = 4.03$	3.21	166 ± 7
ρ ETT (No \vee)	$\epsilon_{v, \varphi_{b1}} = \epsilon_{v, \varphi_{b3}} = 16.64$ $\epsilon_{x_{\Delta, \varphi_{b1}}} = 4.95$ $\epsilon_{x_{\Delta, p}^{l_o}, \varphi_{b3}} = 4.95$ $\epsilon_{x_{\Delta, f}^{l_o}, \varphi_{b2}} = 2.51$	3.41	4846 ± 11
		φ_b (Multilane non-crit.)	
		ρ_{min}	$m \pm \sigma(m)$
TT	— —	20.25	14000
CETT	— —	20.25	1502 ± 24
ρ ETT	— —	20.25	55 ± 2
ρ ETT (No \vee)	— —	20.25	1927 ± 6

TABLE II: An overview of the number of triggered events, minimum robustness and parameters for the found optimal configurations of the TT, CETT and ρ ETT ETT policies. The average number of transmitted measurements and one standard deviation over all simulations for a specific configuration is denoted by $m \pm \sigma(m)$ and ρ_{min} is the minimum robustness over all simulations calculated on the true state. The presence of — || — indicates that the same $\epsilon_{y_i, \varphi}$ parameters are used for the non-critical multilane scenario as for the critical scenario.

sensors. We add additional control behavior to allow the ACC vehicle to change to the fast lane. The vehicles in the fast lane travel at a constant speed and are placed sufficiently far apart to allow for safe overtaking when the lead vehicle brakes.

We construct the PL property $\varphi_b = ((\varphi_{b1}) \vee (\varphi_{b2} \wedge \varphi_{b3}))$ where $\varphi_{b1} = x_{\Delta} > x_{ss}(d_{\varphi}, v, T)$, $\varphi_{b2} = x_{\Delta, f}^{l_o} > x_{ss}(d_{\varphi}, v_f^{l_o}, T)$, $\varphi_{b3} = x_{\Delta, p}^{l_o} > x_{ss}(d_{\varphi}, v, T)$, and the l_o superscript denotes vehicles in the other lane relative to the ACC vehicle and the p and f subscripts denote the vehicles preceding and following the ACC vehicle. We use the following signal-to-property assignments for the inequality properties in φ_b : $\mathbf{y} \sim \varphi_{b1} = \{v, x_{\Delta}\}$, $\mathbf{y} \sim \varphi_{b2} = \{x_{\Delta, f}^{l_o}\}$, $\mathbf{y} \sim \varphi_{b3} = \{x_{\Delta, p}^{l_o}, v\}$. We note that φ_{b2} depends on $v_f^{l_o}$ which is not directly measurable and must be estimated. Thus the ρ ETT policy cannot be used for φ_b . Additionally, we construct an identical property except we replace the \vee operator in φ_b with an \wedge for monitoring. This is to demonstrate what would happen if the ETT relaxation

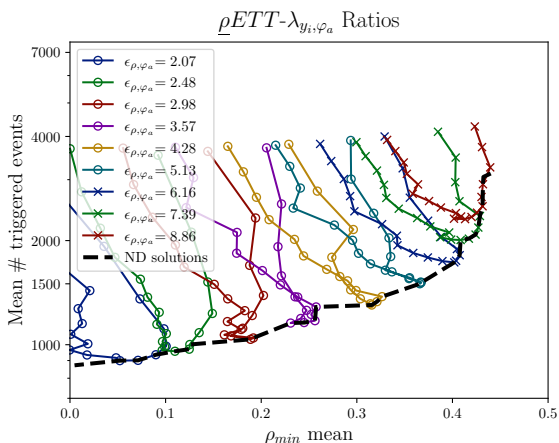


Fig. 7: A plot of the resulting mean ρ_{min} and corresponding mean number of transmitted packet m for different values of $\epsilon_{\rho, \varphi_a}$ and λ_{y_i, φ_a} parameters for ρ_{ETT} . The black ND solutions series corresponds to the non-dominated solutions for the average ρ_{min} values as opposed to Fig. 6 where the overall ρ_{min} was used to find the non-dominated solutions.

mechanism for the \vee operator in Definition 6 was not implemented and we treated the \vee operators as an \wedge . We consider two scenarios: 1) a critical scenario where there is little but enough room to overtake at the right time, and 2) a non-critical scenario where there is plenty of room to overtake.

Parameter identification: Based on the results in the single-lane scenario, we know that $T_s = 0.01s$ is necessary for the TT policy to ensure positive robustness in the most critical scenario. In the case of the multilane scenario, the most critical case is identical to that of the one-lane scenario when there is no room to safely switch to the fast lane. Thus, we conclude that the same sampling frequency is necessary for the multilane scenario.

Next, we consider the CETT and ρ_{ETT} transmission strategies. If there is not enough room to overtake in the fast lane, the ETTs should still ensure positive robustness in the current lane. Thus, the ETTs for the sensors that monitor the speed of the ACC vehicle and the distance between the ACC vehicle and the preceding vehicles in either lane, should use identical parameters to those found for the φ_a safety property in the previous section. However, the property φ_{b_2} in φ_b is a different case, as the estimated speed of the following vehicle is used to determine the safe distance rather than the speed of the ACC vehicle. To determine safe ETT parameters for the sensor measuring the distance to the following vehicle in the fast lane ($x_{\Delta, f}^l$), we initially consider the critical scenario to determine safe parameters. We test a range of different ρ_{ETT} policy parameters for the $x_{\Delta, f}^l$ signal and run 20 simulations for each configuration.

In Table II, we note the minimum robustness and corresponding number of sent measurements in the critical scenario (φ_b) for the best tested configuration for the CETT, ρ_{ETT} , TT and the ρ_{ETT} regulation policy with the \vee replaced by an \wedge operator. The robustness is measured as if the \vee operator were present. The results show that the ρ_{ETT} policy with the

\vee operator triggers **89.1%** and **96.6%** fewer events compared to the CETT and ρ_{ETT} without the \vee operator, respectively. Another notable result is that the ρ_{ETT} without the \vee operator transmits **3.17** times more packets than the CETT approach. Upon investigation, we find that many of the transmitted measurements can be attributed to the vehicles in the fast lane as the robustness of the properties φ_{b_2} and φ_{b_3} is negative for a significant duration of the simulation. This results in the corresponding ETTs being zero for the duration of the negative robustness resulting in many transmitted packets. However, the CETT does not suffer from this problem as the ETTs never reach zero. Based on the results in Fig. 6, we see that once we reach a certain minimum robustness, many additional transmitted measurements will not significantly increase ρ_{min} . Along with the fact that measurement noise is typically present, this may suggest that some minimum ETT > 0 may be a good idea to decrease the number of unnecessary transmitted measurements in the case of low robustness.

Finally, we consider a non-critical multilane scenario where there is plenty of space to overtake the braking vehicle, resulting in an overall larger robustness. The parameters found for the critical case are reused and the results are noted again in Table II. The results for the non-critical multilane scenario show similar results as the critical scenario where the ρ_{ETT} policy transmits **96.3 %** and **97.1%** fewer events compared to the CETT and ρ_{ETT} policy without the \vee operator respectively. The ρ_{ETT} policy triggers significantly fewer events in the non-critical scenario compared to the critical scenario while the number of triggered events for the CETT policy stays almost the same. This result highlights the ability of the ρ_{ETT} mechanism to adapt based on critical and non-critical situations.

V. DISCUSSION

We now discuss our proposed ETT regulation method in light of the obtained numerical results as well as differences and similarities with other approaches in SoTA.

Parameter Tuning and Formal Guarantees: Section IV also demonstrates the parameter tuning procedure used to select constant ETTs or ETT regulation parameters. In the simulated scenarios, the parameters chosen ensure safety. However, these parameters do not provide formal guarantees for a class of systems or scenarios, i.e., they do not provide a guarantee for potentially similar scenarios. For example, if we consider the multilane scenario and the sensor measuring $x_{\Delta, f}^l$, the parameters resulting in triggering the smallest number of events were sufficient to ensure a positive value of ρ_{min} . This is likely due to added acceleration meaning that the estimated robustness of φ_{b_3} is typically larger than the estimated robustness. However, if the drift was in the opposite direction, then the large ETT (or small $\epsilon_{y_i, \varphi}$ value) may not be sufficient to satisfy the requirement. As a result, when using such a numerical parameter tuning method, one has to consider what situations have to be tested to ensure that the chosen parameters ensure safety in all situations that the system will experience, which is beyond the scope of this paper.

A potential way to combat the safety issue described above is to make the control behavior aware of the uncertainty

introduced by the event-triggering threshold. For example, consider the scenario where the ACC vehicle is far away from the safe distance to a preceding vehicle in the slow lane but for some reason desires to change to the fast lane. The property φ_b will then provide high ETTs to the sensors measuring the distance to the vehicles in the fast lane. If the ACC vehicle for some reason desires to switch lanes, it will have an inaccurate estimate of the vehicles in the other lane, which may not guarantee a safe overtaking. Thus, the uncertainty information should be taken into account in the control behavior to ensure safe operation. The reason this is not an issue in the case study, is that the overtaking behavior is designed to only occur when the ACC vehicle is close to the lead vehicle in the slow lane. In this scenario, the low distance to the lead vehicle will not contribute to altering the ETTs related to the distance of the other vehicles in the fast lane through the \vee operator.

To provide formal guarantees, other comparable methods, such as the co-design of a controller and event-triggering mechanism that ensures the stability of a networked control system [30], [5] have been proposed. Such formalisms rely on the ability to formalize a stability condition, which our method does not require as the safety condition is implicitly specified in the optimization problem. Additionally, our method allows us to simultaneously consider performance properties. Although this was not shown in our case study, the overall procedure and concept are the same.

Similarities and Differences with SoTA: Using propositional properties to describe system safety/performance requirements has similarities with other ways of describing systems, such as Fuzzy modeling and control [31]. Fuzzy modeling and control allows the specification of IF-THEN statements to describe control and system behavior which is conceptually similar to how the \rightarrow operator can be used. Several works have been published for event-triggered control of fuzzy systems [32], [33], [34] but they typically only consider behaviors of the form $(\varphi_1 \wedge \varphi_2 \dots) \rightarrow \text{behavior}$. Additionally, many of the formal methods that guarantee safety, do not explore the tradeoff between varying the ETTs for different sensors and its impact on the overall number of transmitted packets.

Connection to Signal Temporal Logic: As mentioned in Section II-D, PL is a subset of STL, which extends propositional logic with temporal operators. For example, the *eventually* (\diamond_I) and *always* (\square_I) operators allow the system to verify that something will eventually become true or is always true within some interval I , respectively. This enables STL to specify properties over constrained time intervals rather than simply stating that something always has to be true. Several results on controller synthesis to ensure satisfaction of STL properties or other similar formalisms have been published (some useful references include [35], [36]) but only very few on event-triggered control [37], [38], [39] exist. Furthermore, most of the event-triggered control publications only consider a subset of STL excluding the \vee operator. To our knowledge, the publication that is closest to our work is [39], which develops an event-triggering strategy that ensures that the event-triggered state trajectory stays within an ϵ -tube of the ideal trajectory given by the continuous state

feedback controller for Temporal Logic over Reals (RTL) [40] formulas. These correspond roughly to unbounded PL formulas but without the definition of numerical satisfaction semantics. Additionally, the event-triggering condition in [39] is evaluated based on the full state and only considers the option of transmitting the full state, rather than measurements of individual signals.

Another similarity with the SoTA is that the optimization problems presented in this paper are always constrained to ensure that the robustness of the property is positive for all time steps. This corresponds to an implicit unbounded STL \square operator. The subset of STL considered in [37], [38], considers bounded \square operators but does not consider disjunctions. Extending our approach to include the STL temporal operators and refining the currently proposed ETT regulation mechanisms will be a core part of our future work.

Assumptions and Impact in Real Systems: The case study makes several other assumptions that need to be handled in a real-world setting. For example, we assumed that the necessary information for evaluating the event-triggering condition is available at the sensor without the need for communication. In a real-world scenario, this assumption does not hold in every case and the required communication could diminish the savings introduced by the ETT. The SOD update error may be a better fit in a real-world scenario as it does not require additional information to predict the value of the measurement. We also note that the ETT refinement as a result of the \vee operator will likely also require additional communication since properties (and thus signals) can affect the ETT of other signals. However, the fact that there is a large gap between the ρ ETT and the constant ETT (namely, the ρ ETT approach triggers between **41.8 - 96.3 %** less events compared to the CETT), provides ample room to address these challenges while maintaining significant gains. Despite many of these potential challenges, the expressiveness of PL has the potential to save communication resources and optimize performance in a variety of scenarios by enabling fine-grained control of the ETT through our proposed method.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented a novel ETT regulation mechanism that infuses ETT regulation with propositional logic. Through the expressiveness of PL, our approach enables accurately defining circumstances in a wide array of problems under which the ETTs are either increased or decreased depending on the safety and/or performance requirements of the system as well as how well these requirements are satisfied at runtime. We explored the intuition and possibilities of our method using a safety-related case study. The case study showed a large potential for reduction in the number of triggered events, e.g. between **41.8 - 96.3 %** fewer events while maintaining similar minimum safety. Assessing the usability of our proposed method in a wide range of systems will be a core part of our future work as this will help us identify potential issues and beneficial improvements. Future work will focus on expanding our approach to include temporal operators from Signal Temporal Logic (STL) in order to capture more

complex properties that have specific timing considerations as well as methods for parameter tuning of the system.

APPENDIX

A. Proof of Lemma 1

Proof. The scalar α can be represented as the degenerate interval $A = [\alpha, \alpha]$. Multiplying X and A , according to eq. (16), we have: $A \cdot X = [\min S, \max S]$ where $S = \{\underline{AX}, \overline{AX}, \underline{AX}, \overline{AX}\}$. Since $\bar{A} = \underline{A}$, then $\underline{AX} = \overline{AX}$ and $\overline{AX} = \underline{AX}$. Since $\alpha < 0$ and $\bar{X} > \underline{X}$, consequently $\alpha\bar{X} < \alpha\underline{X}$, resulting in $\min S = \alpha\bar{X}$ and $\max S = \alpha\underline{X}$ and the interval $A \cdot X = [\alpha\bar{X}, \alpha\underline{X}]$. \square

B. Proof of Lemma 2

Proof. We show the proof for the case $\alpha_X > 0$ and $\alpha_Y < 0$. The proof for the remaining cases ($\alpha_X > 0$ and $\alpha_Y > 0$, $\alpha_X < 0$ and $\alpha_Y > 0$, $\alpha_X < 0$ and $\alpha_Y < 0$) can be obtained by following the same approach. Using the result of Lemma 1:

$$\begin{aligned} & w(\alpha_X X(x, \delta_x) + \alpha_Y Y(y, \delta_y)) \\ &= w([\alpha_X(x - \delta_x), \alpha_X(x + \delta_x)] + [-|\alpha_Y|(y + \delta_y), -|\alpha_Y|(y - \delta_y)]) \\ &= w([\alpha_X(x - \delta_x) - |\alpha_Y|(y + \delta_y), \alpha_X(x + \delta_x) - |\alpha_Y|(y - \delta_y)]) \\ &= \alpha_X(x + \delta_x) - |\alpha_Y|(y - \delta_y) - (\alpha_X(x - \delta_x) - |\alpha_Y|(y + \delta_y)) \\ &= 2\alpha_X\delta_x + 2|\alpha_Y|\delta_y = 2|\alpha_X|\delta_x + 2|\alpha_Y|\delta_y. \end{aligned}$$

\square

REFERENCES

- [1] W. Chen, D. Shi, J. Wang, and L. Shi, "Event-Triggered State Estimation: Experimental Performance Assessment and Comparative Study," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 5, pp. 1865–1872, Sep. 2017.
- [2] K.-E. Årzén, "A simple event-based PID controller," *IFAC Proceedings Volumes*, vol. 32, no. 2, pp. 8687–8692, Jul. 1999.
- [3] K. Johan Åström and B. Bernhardsson, "Comparison of periodic and event based sampling for first-order stochastic systems," *IFAC Proceedings Volumes*, vol. 32, no. 2, pp. 5006–5011, Jul. 1999.
- [4] L. Ding, Q.-L. Han, X. Ge, and X.-M. Zhang, "An Overview of Recent Advances in Event-Triggered Consensus of Multiagent Systems," *IEEE Transactions on Cybernetics*, vol. 48, no. 4, pp. 1110–1123, Apr. 2018.
- [5] X.-M. Zhang, Q.-L. Han, X. Ge, D. Ding, L. Ding, D. Yue, and C. Peng, "Networked control systems: A survey of trends and techniques," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 1, pp. 1–17, Jan. 2020.
- [6] X.-C. Jia, "Resource-efficient and secure distributed state estimation over wireless sensor networks: A survey," *International Journal of Systems Science*, vol. 52, no. 16, pp. 3368–3389, Dec. 2021.
- [7] X. Li and G. Hao, "Event-Triggered Kalman Filter and Its Performance Analysis," *Sensors*, vol. 23, no. 4, p. 2202, Jan. 2023.
- [8] X.-M. Zhang and Q.-L. Han, "Event-based H_∞ filtering for sampled-data systems," *Automatica*, vol. 51, pp. 55–69, Jan. 2015.
- [9] G. Wei, S. Liu, L. Wang, and Y. Wang, "Event-based distributed set-membership filtering for a class of time-varying non-linear systems over sensor networks with saturation effects," *International Journal of General Systems*, vol. 45, no. 5, pp. 532–547, Jul. 2016.
- [10] H. K. Büning and T. Lettmann, *Propositional Logic: Deduction and Algorithms*. Cambridge University Press, Aug. 1999.
- [11] X. Ge, Q.-L. Han, L. Ding, Y.-L. Wang, and X.-M. Zhang, "Dynamic Event-Triggered Distributed Coordination Control and its Applications: A Survey of Trends and Techniques," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 9, pp. 3112–3125, Sep. 2020.
- [12] X. Ge, Q.-L. Han, X.-M. Zhang, and D. Ding, "Dynamic Event-triggered Control and Estimation: A Survey," *International Journal of Automation and Computing*, vol. 18, no. 6, pp. 857–886, Dec. 2021.
- [13] X. Ge, Q.-L. Han, X.-M. Zhang, L. Ding, and F. Yang, "Distributed Event-Triggered Estimation Over Sensor Networks: A Survey," *IEEE Transactions on Cybernetics*, vol. 50, no. 3, pp. 1306–1320, Mar. 2020.
- [14] X.-M. Zhang, Q.-L. Han, and B.-L. Zhang, "An Overview and Deep Investigation on Sampled-Data-Based Event-Triggered Control and Filtering for Networked Systems," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 1, pp. 4–16, Feb. 2017.
- [15] M. Miskowicz, "Send-On-Delta Concept: An Event-Based Data Reporting Strategy," *Sensors*, vol. 6, no. 1, pp. 49–63, Jan. 2006.
- [16] G. S. Seyboth, D. V. Dimarogonas, and K. H. Johansson, "Event-based broadcasting for multi-agent average consensus," *Automatica*, vol. 49, no. 1, pp. 245–252, Jan. 2013.
- [17] W. Hu, L. Liu, and G. Feng, "Consensus of Linear Multi-Agent Systems by Distributed Event-Triggered Strategy," *IEEE Transactions on Cybernetics*, vol. 46, no. 1, pp. 148–157, Jan. 2016.
- [18] H. Li, J. Luo, H. Ma, and Q. Zhou, "Observer-Based Event-Triggered Iterative Learning Consensus for Locally Lipschitz Nonlinear MASs," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 16, no. 1, pp. 46–56, Feb. 2024.
- [19] X. Ge, S. Xiao, Q.-L. Han, X.-M. Zhang, and D. Ding, "Dynamic Event-Triggered Scheduling and Platooning Control Co-Design for Automated Vehicles Over Vehicular Ad-Hoc Networks," *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 1, pp. 31–46, Jan. 2022.
- [20] X.-M. Zhang, Q.-L. Han, X. Ge, D. Ding, L. Ding, D. Yue, and C. Peng, "Networked control systems: A survey of trends and techniques," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 1, pp. 1–17, Jan. 2020.
- [21] D. Shi, L. Shi, and T. Chen, *Event-Based State Estimation*, ser. Studies in Systems, Decision and Control. Cham: Springer International Publishing, 2016, vol. 41.
- [22] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. Jordan, and S. Sastry, "Kalman Filtering With Intermittent Observations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1453–1464, Sep. 2004.
- [23] D. Shi, T. Chen, and L. Shi, "An event-triggered approach to state estimation with multiple point- and set-valued measurements," *Automatica*, vol. 50, no. 6, pp. 1641–1648, Jun. 2014.
- [24] Y. S. Suh, V. H. Nguyen, and Y. S. Ro, "Modified Kalman filter for networked monitoring systems employing a send-on-delta method," *Automatica*, vol. 43, no. 2, pp. 332–338, Feb. 2007.
- [25] O. Maler and D. Nickovic, "Monitoring Temporal Properties of Continuous Signals," in *Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems*, ser. Lecture Notes in Computer Science, Y. Lakhnech and S. Yovine, Eds. Berlin, Heidelberg: Springer, 2004, pp. 152–166.
- [26] R. E. Moore, R. B. Kearfott, and M. J. Cloud, *Introduction to Interval Analysis*. Society for Industrial and Applied Mathematics, Jan. 2009.
- [27] W. Hu, L. Liu, and G. Feng, "Consensus of Linear Multi-Agent Systems by Distributed Event-Triggered Strategy," *IEEE Transactions on Cybernetics*, vol. 46, no. 1, pp. 148–157, Jan. 2016.
- [28] M. Milanese and A. Vicino, "Optimal estimation theory for dynamic systems with set membership uncertainty: An overview," *Automatica*, vol. 27, no. 6, pp. 997–1009, Nov. 1991.
- [29] M. Treiber, A. Hennecke, and D. Helbing, "Congested Traffic States in Empirical Observations and Microscopic Simulations," *Physical Review E*, vol. 62, no. 2, pp. 1805–1824, Aug. 2000.
- [30] X.-M. Zhang, Q.-L. Han, and X. Yu, "Survey on Recent Advances in Networked Control Systems," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 5, pp. 1740–1752, Oct. 2016.
- [31] R. R. Yager and D. P. Filev, *Essentials of Fuzzy Modeling and Control*. USA: Wiley-Interscience, Jun. 1994.
- [32] Y. Pan and G.-H. Yang, "Event-triggered fuzzy control for nonlinear networked control systems," *Fuzzy Sets and Systems*, vol. 329, pp. 91–107, Dec. 2017.
- [33] X. Su, F. Xia, J. Liu, and L. Wu, "Event-triggered fuzzy control of nonlinear systems with its application to inverted pendulum systems," *Automatica*, vol. 94, pp. 236–248, Aug. 2018.
- [34] Y. Pan, Y. Wu, and H.-K. Lam, "Security-Based Fuzzy Control for Nonlinear Networked Control Systems With DoS Attacks via a Resilient Event-Triggered Scheme," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 10, pp. 4359–4368, Oct. 2022.
- [35] C. Belta and S. Sadraddini, "Formal Methods for Control Synthesis: An Optimization Perspective," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 2, no. 1, pp. 115–140, 2019.
- [36] V. Raman, A. Donzé, D. Sadigh, R. M. Murray, and S. A. Seshia, "Reactive synthesis from signal temporal logic specifications," in *Proceedings of the 18th International Conference on Hybrid Systems: Computation*

- and Control*, ser. HSCC '15. New York, NY, USA: Association for Computing Machinery, Apr. 2015, pp. 239–248.
- [37] L. Lindemann, D. Maity, J. S. Baras, and D. V. Dimarogonas, “Event-triggered Feedback Control for Signal Temporal Logic Tasks,” in *2018 IEEE Conference on Decision and Control (CDC)*. Miami Beach, FL: IEEE, Dec. 2018, pp. 146–151.
- [38] L. Long and G. Gao, “Dynamic event-triggered prescribed performance control for nonlinear systems with signal temporal logic,” *International Journal of Robust and Nonlinear Control*, p. rnc.7226, Feb. 2024.
- [39] D. Maity and J. S. Baras, “Event-Triggered Controller Synthesis for Dynamical Systems with Temporal Logic Constraints,” in *2018 Annual American Control Conference (ACC)*, Jun. 2018, pp. 1184–1189.
- [40] M. Reynolds, “The complexity of temporal logic over the reals,” *Annals of Pure and Applied Logic*, vol. 161, no. 8, pp. 1063–1096, May 2010.