Resilience to Non-Compliance in Coupled Cooperating Systems

Brooks A. Butler, *Member, IEEE* and Philip E. Paré, *Senior Member, IEEE*

***Abstract*— This letter explores the implementation of a safe control law for systems of dynamically coupled cooperating agents. Under a CBF-based collaborative safety framework, we examine how the maximum safety capability for a given agent, which is computed using a collaborative safety condition, influences safety requests made to neighbors. We provide conditions under which neighbors may be resilient to non-compliance of neighbors to safety requests, and compute an upper bound for the total amount of non-compliance an agent is resilient to, given its 1-hop neighborhood state and knowledge of the network dynamics. We then illustrate our results via simulations of a networked susceptible-infected-susceptible (SIS) epidemic model.**

***Index Terms*— Collaborative control, network control, safety-critical control**

I. Introduction

ENSURING safety for complex networked systems is a challenging problem across many applications. Some examples of critical networked system applications include smart grid management [1], uncrewed aerial drone swarms [2], [3], multi-agent robot systems [4], and the mitigation of epidemic spreading processes [5]. While much of the foundational work on safety for dynamical systems via set invariance was performed by Nagumo during the 1940s [6], the study of safety-critical control has seen a significant resurgence in recent years, due largely to the introduction and refinement of *control barrier functions* (CBFs) [7], [8]. Of particular interest for this work are scenarios where agents in networked systems may be treated as independent entities with individual safety requirements. This dynamically coupled environment motivates the need for collaborative frameworks that can facilitate cooperation between neighbors and enable the collective safety of all agents [9]–[14].

Resilience for networked systems is crucial to achieving reliable performance [15]–[18]. In this work, we define resilience similarly to [15], which is the ability of the network to operate reliably even after experiencing some failures. Note that resilience is different from the property of robustness, which is the ability of systems to reject disturbances such as noise. For example, a robust system may be able to handle uncertainty or noise in the system states; however, a resilient system should be able to adapt to fundamental changes in the system's capability, such as the loss of an actuator or sudden change in control constraints due to loss of resources. In multi-agent systems, compliance is used to describe an agent's behavior with respect to the adherence to group norms [19]; however, in this work, we consider compliance explicitly as an agent's ability, or willingness, to fulfill requests by neighbors in a collaborative setting.

In our previous work, we developed a CBF-based safety-filter control law that defines a collaborative safety framework for cooperating coupled agents [20] and applied this framework to a formation control problem with obstacle avoidance [21]. In this work, we examine the case where, under the previously developed collaborative framework, agents may be unable to fulfill their commitments to maintain their neighbors' safety requirements. Thus, we investigate how resilient each agent is to this possible neighbor failure, which is a first step towards identifying factors that could lead to cascading failures across the network [22]. In other words, we investigate the following question: *How resilient is each agent to non-compliant neighbors?*

Thus, in this letter, we make the following contributions:

- Under the formulation of our collaborative safety framework [20], we show how the computation of the maximum safety capability for a given agent influences its safety requests made to neighbors.
- We define a metric for non-compliance and provide an upper bound of the total amount of non-compliance an agent is resilient to given its 1-hop neighborhood state and knowledge of networked dynamics.

II. Preliminaries

In this section, we define the notation used in this paper, introduce preliminaries for networked dynamic systems, and discuss safety definitions for networked systems.

A. Notation

Let Int$\mathcal{C}$, $\partial\mathcal{C}$, $|\mathcal{C}|$ denote the interior, boundary, and cardinality of the set $\mathcal{C}$, respectively. $\mathbb{R}$ and $\mathbb{N}$ are the sets of real numbers and positive integers, respectively. Let D^r denote the set of functions r-times continuously differentiable in all arguments, and $\mathcal{K}$ the set of class-$\mathcal{K}$ functions. We define $[n] \subset \mathbb{N}$ to be a set of indices $\{1, 2, \dots, n\}$. We define the Lie derivative of the function $h : \mathbb{R}^N \to \mathbb{R}$ with respect to the vector field generated by $f : \mathbb{R}^N \to \mathbb{R}^N$ as

$$\mathcal{L}_f h(x) = \frac{\partial h(x)}{\partial x} f(x). \tag{1}$$

We define higher-order Lie derivatives with respect to the same vector field f with a recursive formula [23], for $k > 1$, as

$$\mathcal{L}_f^k h(x) = \frac{\partial \mathcal{L}_f^{k-1} h(x)}{\partial x} f(x). \tag{2}$$

*Brooks A. Butler is with the Department of Electrical Engineering and Computer Science at the University of California, Irvine and Philip E. Paré is with the Elmore Family School of Electrical and Computer Engineering at Purdue University. Emails: bbutler2@uci.edu and philpare@purdue.edu. This work was partially funded by Purdue's Elmore Center for Uncrewed Aircraft Systems and the National Science Foundation, grant NSF-ECCS #2238388.

B. Networked Dynamic System Model

We define a networked system using a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of $n = |\mathcal{V}|$ nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges. Let $\mathcal{N}_i^+$ be the set of all neighbors with an incoming connection to node i, where

$$\mathcal{N}_i^+ = \{j \in [n] \setminus \{i\} : (i, j) \in \mathcal{E}\}. \tag{3}$$

Similarly, all nodes with an outgoing connection from i to j are given by

$$\mathcal{N}_i^- = \{j \in [n] \setminus \{i\} : (j, i) \in \mathcal{E}\}, \tag{4}$$

with the complete set of neighboring nodes given by

$$\mathcal{N}_i = \mathcal{N}_i^+ \cup \mathcal{N}_i^-. \tag{5}$$

Further, we define the state vector for each node as $x_i \in \mathbb{R}^{N_i}$, with $N = \sum_{i \in [n]} N_i$ being the state dimension of the entire system, $N_i^+ = \sum_{j \in \mathcal{N}_i^+} N_j$ the combined dimension of incoming neighbor states, and $x_{\mathcal{N}_i^+} \in \mathbb{R}^{N_i^+}$ denoting the combined state vector of all incoming neighbors. Then, for each node $i \in [n]$, we describe its state dynamics, which are potentially nonlinear, time-invariant, and control-affine, as

$$\dot{x}_i = f_i(x_i, x_{\mathcal{N}_i^+}) + g_i(x_i) u_i, \tag{6}$$

where $f_i : \mathbb{R}^{N_i + N_i^+} \to \mathbb{R}^{N_i}$ and $g_i : \mathbb{R}^{N_i} \to \mathbb{R}^{N_i} \times \mathbb{R}^{M_i}$ are locally Lipschitz for all $i \in [n]$, and $u_i \in \mathcal{U}_i \subset \mathbb{R}^{M_i}$. For notational compactness, given a node $i \in [n]$, we collect the 1-hop neighborhood state as $\mathbf{x}_i = (x_i, x_{\mathcal{N}_i^+})$ and the 2-hop neighborhood state as $\mathbf{x}_i^+ = (x_i, x_{\mathcal{N}_i^+}, x_{\mathcal{N}_j^+} \; \forall j \in \mathcal{N}_i^+)$.

C. Safety Definitions

We define a node-level safety constraint for node $i \in [n]$ with the set

$$\mathcal{C}_i = \left\{ x_i \in \mathbb{R}^{N_i} : h_i(x_i) \geq 0 \right\}, \tag{7}$$

where $h_i \in D^r, r \geq 1$ and $h_i : \mathbb{R}^{N_i} \to \mathbb{R}$ is a function whose zero-super-level set defines the region which node $i \in [n]$ considers to be safe (i.e., if $h_i(x_i) < 0$, then node i is no longer safe).

III. Safety in Networked Dynamic Systems

One challenge that comes from the definition of node-level safety constraints (defined in Section II-C), which depend only on the state of the given node, is that, when evaluating the derivative of $h_i(x_i)$, we must incorporate the network influence of its neighbors $j \in \mathcal{N}_i^+$, where we define the derivative of h_i as

$$\dot{h}_i(\mathbf{x}_i, u_i) = \mathcal{L}_{f_i} h_i(\mathbf{x}_i) + \mathcal{L}_{g_i} h_i(x_i) u_i.$$

Notice that, in this case, we can define the node-level constraint function as the mapping $h_i : \mathbb{R}^{N_i} \to \mathbb{R}$; however, the dynamics of node i, whose dynamics are a function of all neighboring nodes in $\mathcal{N}_i^+$, require that $\dot{h}_i : \mathbb{R}^{N_i + N_i^+} \to \mathbb{R}$. Therefore, it is possible for neighboring states $x_{\mathcal{N}_i}$, whose dynamics are controlled by each neighbor $j \in \mathcal{N}_i$ through $u_j \in \mathbb{R}^{M_j}$, to influence the safety of node i.

Using principles from *high-order barrier functions* [24], [25], we can analyze the effect of the 1-hop neighborhood dynamics on the safety of node i by computing the second-order derivative of h_i, which can be expressed as

$$\begin{aligned} \ddot{h}_i(\mathbf{x}_i^+, u_i, u_{\mathcal{N}_i^+}, \dot{u}_i) = \sum_{j \in \mathcal{N}_i^+} &\left[\mathcal{L}_{f_j} \mathcal{L}_{f_i} h_i(\mathbf{x}_i^+) + \mathcal{L}_{g_j} \mathcal{L}_{f_i} h_i(\mathbf{x}_i) u_j \right] \\ &+ \mathcal{L}_{f_i}^2 h_i(\mathbf{x}_i) + u_i^\top \mathcal{L}_{g_i}^2 h_i(x_i) u_i + \mathcal{L}_{g_i} h_i(x_i) \dot{u}_i \\ &+ \left(\mathcal{L}_{f_i} \mathcal{L}_{g_i} h_i(\mathbf{x}_i)^\top + \mathcal{L}_{g_i} \mathcal{L}_{f_i} h_i(\mathbf{x}_i) \right) u_i. \end{aligned} \tag{8}$$

Notice that the dynamics of each neighbor $j \in \mathcal{N}_i^+$ and $\dot{u}_i$ appear in the second-order differential expression of h_i. To assist in our analysis of the high-order dynamics of h_i, we make the following assumption.

Assumption 1. *For a given node $i \in [n]$, let $\dot{u}_i := d(u_i)$, where $d(u_i) : \mathbb{R}^{M_i} \to \mathbb{R}^{M_i}$ is locally Lipschitz.*

While obtaining a closed-form solution for $\dot{u}_i$ may be challenging in some applications, in practice, $d(u_i)$ may be approximated using discrete-time methods. Given the structure of (8), which includes the 1-hop neighborhood controlled dynamics, we define a series of functions under Assumption 1

$$\begin{aligned} \psi_i^0(x_i) &:= h_i(x_i) \\ \psi_i^1(\mathbf{x}_i, u_i) &:= \dot{\psi}_i^0(\mathbf{x}_i, u_i) + \eta_i(\psi_i^0(x_i)) \\ \psi_i^2(\mathbf{x}_i^+, u_i, u_{\mathcal{N}_i^+}) &:= \dot{\psi}_i^1(\mathbf{x}_i^+, u_i, u_{\mathcal{N}_i^+}) + \kappa_i(\psi_i^1(\mathbf{x}_i, u_i)), \end{aligned} \tag{9}$$

where η_i, κ_i are class-$\mathcal{K}$ functions. We can also express $\psi_i^2(\mathbf{x}_i^+, u_i, u_{\mathcal{N}_i^+})$ in terms of h_i as

$$\begin{aligned} \psi_i^2(\mathbf{x}_i^+, u_i, u_{\mathcal{N}_i^+}) = \ddot{h}_i(\mathbf{x}_i^+, u_i, u_{\mathcal{N}_i^+}) + \dot{\eta}_i(h_i(x_i), \mathbf{x}_i, u_i) \\ + \kappa_i\big(\dot{h}_i(\mathbf{x}_i, u_i) + \eta_i(h_i(x_i))\big). \end{aligned} \tag{10}$$

Further, we may rewrite (10) by collecting the terms that are independent of u_j as

$$\psi_i^2(\mathbf{x}_i^+, u_i, u_{\mathcal{N}_i^+}) = \sum_{j \in \mathcal{N}_i^+} a_{ij}(\mathbf{x}_i) u_j + c_i(\mathbf{x}_i^+, u_i), \tag{11}$$

where

$$a_{ij}(\mathbf{x}_i) = \mathcal{L}_{g_j} \mathcal{L}_{f_i} h_i(\mathbf{x}_i) \tag{12}$$

and

$$\begin{aligned} c_i(\mathbf{x}_i^+, u_i) = \sum_{j \in \mathcal{N}_i^+} &\mathcal{L}_{f_j} \mathcal{L}_{f_i} h_i(\mathbf{x}_i^+) + \mathcal{L}_{f_i}^2 h_i(\mathbf{x}_i) + u_i^\top \mathcal{L}_{g_i}^2 h_i(x_i) u_i \\ &+ \left(\mathcal{L}_{f_i} \mathcal{L}_{g_i} h_i(\mathbf{x}_i)^\top + \mathcal{L}_{g_i} \mathcal{L}_{f_i} h_i(\mathbf{x}_i) \right) u_i + \mathcal{L}_{g_i} h_i(x_i) d(u_i) \\ &+ \dot{\eta}_i(h_i(x_i), \mathbf{x}_i, u_i) + \kappa_i\big(\dot{h}_i(\mathbf{x}_i, u_i) + \eta_i(h_i(x_i))\big). \end{aligned} \tag{13}$$

In this form, we may consider $a_{ij}(\mathbf{x}_i) \in \mathbb{R}^{M_j}$ to be the effect that node $j \in \mathcal{N}_i^+$ has on the safety of node $i \in [n]$ via its own control inputs u_j, and $c_i(\mathbf{x}_i^+, u_i)$ is the effect that node $i \in [n]$ has on its own safety combined with the uncontrolled system dynamics. The functions in (9) in turn define the constraint sets

$$\begin{aligned} \mathcal{C}_i^1 &= \{\mathbf{x}_i \in \mathbb{R}^{N_i + N_i^+} : \psi_i^0(x_i) \geq 0\} \\ \mathcal{C}_i^2 &= \{\mathbf{x}_i \in \mathbb{R}^{N_i + N_i^+} : \exists u_i \in \mathcal{U}_i \text{ s.t. } \psi_i^1(\mathbf{x}_i, u_i) \geq 0\}. \end{aligned} \tag{14}$$

We define a *collaborative control barrier function* for node i that takes into account the control actions of its incoming neighbors $j \in \mathcal{N}_i^+$.

Definition 1. *Let $\mathcal{C}_i^1$ and $\mathcal{C}_i^2$ be defined by* (9) *and* (14)*, under Assumption 1. We have that h_i is a **collaborative control barrier function** (CCBF) for node $i \in [n]$ if $h_i \in C^2$ and $\forall \mathbf{x}_i \in \mathcal{C}_i^1 \cap \mathcal{C}_i^2$ there exists $(u_i, u_{\mathcal{N}_i^+}) \in \mathcal{U}_i \times \mathcal{U}_{\mathcal{N}_i^+}$ such that*

$$\psi_i^2(\mathbf{x}_i^+, u_i, u_{\mathcal{N}_i^+}) \geq 0, \tag{15}$$

where η_i, κ_i are class-$\mathcal{K}$ functions and $\eta_i \in D^r$, with $r \geq 1$.

In [20], we prove the forward invariance of $\mathcal{C}_i^1 \cap \mathcal{C}_i^2$ given the existence of a CCBF and use this property to construct a decentralized algorithm for *collaborative safety* among cooperating network agents. A high-level pseudo-code description of this procedure is given in Algorithm 1, where, for a given agent $i \in [n]$, safety commitments $\bar{c}_{ij}$ and $\bar{c}_{ki}$ are initialized for incoming neighbors $j \in \mathcal{N}_i^+$ and outgoing neighbors $k \in \mathcal{N}_i^-$, respectively, as well as the set of allowable actions for agent i, $\overline{\mathcal{U}}_i$. Then, for each round of collaboration τ_i, agent i computes its maximum safety capability as

$$\bar{c}_i = \max_{u_i \in \mathcal{U}_i} c_i(\mathbf{x}_i^+, u_i), \tag{16}$$

which is used as input for the Collaborate function on Line 5, defined in [20], to negotiate safety requests between neighbors $\mathcal{N}_i$ using the collaborative safety condition as defined in (15). After each round of collaboration, a safety deficit/surplus δ_i is returned along with the updated set of allowed actions for agent i that can satisfy the safety requirements of both agent i and its neighbors, as well as the current safety commitments for all neighbors. If $\delta_i \geq 0$, indicating their safety requirement is met, then Algorithm 1 returns the set of feasibly safe actions $\overline{\mathcal{U}}_i$ for agent i and its neighbors $\mathcal{N}_i^-$. In [20], we provide conditions under which Algorithm 1 will converge asymptotically to at least one safe action for all agents, if it exists. In this paper, we explore the scenario where there is potential error, or non-compliance, in the safety commitments $\bar{c}_{ij}$ given by each neighbor $j \in \mathcal{N}_i^+$ and how resilient each agent is to the collective non-compliance of all neighbors.

Algorithm 1 Collaborative Safety

1: **Initialize:**
  $i \leftarrow i_0$; $\overline{\mathcal{U}}_i \leftarrow \mathcal{U}_i$; $\tau_i \leftarrow 0$
  $\bar{c}_{ij} \leftarrow 0, \forall j \in \mathcal{N}_i^+$; $\bar{c}_{ki} \leftarrow 0, \forall k \in \mathcal{N}_i^-$
2: **repeat**
3:   $\tau_i \leftarrow \tau_i + 1$
4:   $\bar{c}_i \leftarrow \max_{u_i \in \overline{\mathcal{U}}_i} c_i(\mathbf{x}_i^+, u_i)$
5:   $\delta_i, \overline{\mathcal{U}}_i, \{\bar{c}_{ij}\}_{j \in \mathcal{N}_i^+}, \{\bar{c}_{ki}\}_{k \in \mathcal{N}_i^-}$
    $\leftarrow$ Collaborate$\left(\bar{c}_i, \overline{\mathcal{U}}_i, \{\bar{c}_{ij}\}_{j \in \mathcal{N}_i^+}, \{\bar{c}_{ki}\}_{k \in \mathcal{N}_i^-}\right)$ [20]
6: **until** $\delta_i \geq 0$
7: **return** $\overline{\mathcal{U}}_i$

IV. Resilience to Non-Compliance

In this section, we explore properties of resilience in the context of collaborative safety via Algorithm 1. For our discussion, we consider the choice of class-$\mathcal{K}$ functions $\eta_i(\cdot)$ and $\kappa_i(\cdot)$ as a part of $\psi_i^1(\mathbf{x}_i, u_i)$ and $\psi_i^2(\mathbf{x}_i^+, u_i, u_{\mathcal{N}_i^+})$ and their effect on the behavior of neighbors with respect to communicating safety needs to neighbors and choosing actions that guarantee individual node safety. To simplify our discussion, we consider the set of linear class-$\mathcal{K}$ functions.

Assumption 2. *Let $\eta_i(z) := \eta_i z$ and $\kappa_i(z) := \kappa_i z$, where $z \in \mathbb{R}^{N_i}$ and $\eta_i, \kappa_i \in \mathbb{R}_{\geq 0}$.*

The selection of η_i and κ_i may be used to determine the qualitative behavior of a given node, where large η_i (κ_i) will prevent the first (second) derivative safety condition in (9) from activating until x_i approaches much closer to the boundary of $\mathcal{C}_i^1$ ($\mathcal{C}_i^2$). In other words, we can use η_i and κ_i to tune how conservative node i is in avoiding the boundary of its safety constraints. Additionally, we make the following assumption on the control constraints for node $i \in [n]$.

Assumption 3. *Let $\mathcal{U}_i \subset \mathbb{R}^{M_i}$ be a closed set.*

Finally, we define a first-order approximation of $d(u_i)$ to be used in the computation of the safety capability of node i as

$$d(u_i) \approx \tilde{d}(u_i) = \frac{u_i^0 - u_i}{\Delta_t}, \tag{17}$$

where u_i^0 is the last chosen control action for node i and $\Delta_t > 0$ is the time difference between the last action and the current action u_i. Note that a key step in collaborative safety via Algorithm 1 is the evaluation of the maximum capability of node $i \in [n]$ through (13) and (16). Under Assumption 2 and (17), we may express (13) as

$$\begin{aligned} c_i(\mathbf{x}_i^+, u_i, \nu_i) = &\sum_{j \in \mathcal{N}_i^+} \mathcal{L}_{f_j}\mathcal{L}_{f_i} h_i(\mathbf{x}_i^+) + \mathcal{L}_{f_i}^2 h_i(\mathbf{x}_i) + u_i^\top \mathcal{L}_{g_i}^2 h_i(x_i) u_i \\ &+ \kappa_i \eta_i h_i(x_i) + \nu_i \mathcal{L}_{f_i} h_i(\mathbf{x}_i) + \mathcal{L}_{g_i} h_i(x_i)\tilde{d}(u_i) \\ &+ \left(\mathcal{L}_{f_i}\mathcal{L}_{g_i} h_i(\mathbf{x}_i)^\top + \mathcal{L}_{g_i}\mathcal{L}_{f_i} h_i(\mathbf{x}_i) + \nu_i \mathcal{L}_{g_i} h_i(\mathbf{x}_i)\right) u_i, \end{aligned}$$

where $\nu_i = \eta_i + \kappa_i$. Thus, solving (16) is equivalent to solving the constrained quadratic program

$$\max_{u_i \in \overline{\mathcal{U}}_i} \quad u_i^\top Q_i(x_i) u_i + [q_i(x) + \nu_i \mathcal{L}_{g_i} h_i(x_i)] u_i + \mathcal{L}_{g_i} h_i(x_i)\tilde{d}(u_i), \tag{18}$$

with

$$Q_i(x_i) = \mathcal{L}_{g_i}^2 h_i(x_i)$$

and

$$q_i(\mathbf{x}_i) = \mathcal{L}_{f_i}\mathcal{L}_{g_i} h_i(\mathbf{x}_i)^\top + \mathcal{L}_{g_i}\mathcal{L}_{f_i} h_i(\mathbf{x}_i).$$

Notice, however, that this maximization problem may not necessarily represent the true maximum capability of node i in satisfying its own safety conditions depending on the properties of $Q_i(x_i) \in \mathbb{R}^{M_i \times M_i}$, i.e., when $Q_i(x_i)$ is negative definite, positive definite, or indefinite. In other words, we must also consider the maximization of $\psi_i^1(\mathbf{x}_i, u_i)$ with respect to its local control input

$$\max_{u_i \in \overline{\mathcal{U}}_i} \quad \mathcal{L}_{g_i} h_i(x_i) u_i. \tag{19}$$

We also consider the following assumption on $\mathcal{L}_{g_i} h_i(x_i)$.

Assumption 4. *For a given $x_i \in \mathbb{R}^{N_i}$, let all entries of $\mathcal{L}_{g_i} h_i(x_i) \in \mathbb{R}^{M_i}$ be non-zero.*

In other words, assume that each control input for node i has some effect on the safety dynamics of h_i. Using the above assumptions, we derive the following results.

Lemma 1. *Given Assumption 3, $\overline{\mathcal{U}}_i \neq \emptyset$, and $|\mathcal{L}_{g_i} h_i(x_i)| > 0$ for some $x_i \in \mathbb{R}^{N_i}$,* (19) *must have a maximal solution $u_i^* \in \partial\overline{\mathcal{U}}_i$, with $u_i^* = \arg\max_{u_i \in \overline{\mathcal{U}}_i} \mathcal{L}_{g_i} h_i(x_i) u_i$. Further, if Assumption 4 holds, then u_i^* is unique.*

Proof. By [20, Algorithm 3], we have that $\overline{\mathcal{U}}_i$ is computed by taking the intersection of $\mathcal{U}_i$ with $\overline{\mathcal{U}}_{\mathcal{N}_i^-} = \bigcap_{k \in \mathcal{N}_i^-} \overline{\mathcal{U}}_{ki}$, where

$$\overline{\mathcal{U}}_{ki} = \{u_i \in \mathbb{R}^{M_i} : a_{ki}(\mathbf{x}_k) u_i + \bar{c}_{ki} + \delta_{ki} \geq 0\}.$$

Thus, since $\overline{\mathcal{U}}_{\mathcal{N}_i^-}$ is the intersection of closed half-spaces, and $\mathcal{U}_i$ is closed by Assumption 3, we have that $\overline{\mathcal{U}}_i = \mathcal{U}_i \cap \overline{\mathcal{U}}_{\mathcal{N}_i^-}$ must also be closed. Thus, the expression $\mathcal{L}_{g_i} h_i(x_i) u_i$ evaluated at some $x_i \in \mathbb{R}^{N_i}$ describes a closed hyperplane in $\mathbb{R}^{M_i}$ with at least one non-zero gradient. Therefore, if we define $u_i^* = [u_i^{1*}, \dots u_i^{M_i *}]^\top$, $\mathcal{L}_{g_i} h_i(x_i) = [\mathcal{L}_{g_i} h_i(x_i)^1, \dots, \mathcal{L}_{g_i} h_i(x_i)^{M_i}]$, and $\overline{\mathcal{U}}_i = \overline{\mathcal{U}}_i^1 \times \cdots \times \overline{\mathcal{U}}_i^{M_i} \neq \emptyset$, then, for each $m \in [M_i]$,

$$u_i^{m*} = \arg\max_{u_i^m \in \partial\overline{\mathcal{U}}_i^m} \mathcal{L}_{g_i} h_i(x_i)^m u_i^m. \tag{20}$$

Further, if $|\mathcal{L}_{g_i} h_i(x_i)^m| > 0$ for all $m \in [M_i]$, i.e., if Assumption 4 holds, then u_i^{m*} is unique for $m \in [M_i]$ since (20) is maximizing linear variables with non-zero gradients constrained by the closed set $\overline{\mathcal{U}}_i \subset \mathbb{R}^{M_i}$. □

Since $\mathcal{L}_{g_i} h_i(x_i) u_i$ is a term in (18), we gain the following insight on the relationship between (18) and (19).

Lemma 2. *Let Assumptions 1-4 hold for some $x_i \in \mathbb{R}^{N_i}$ and $x_{\mathcal{N}_i^+} \in \mathbb{R}^{N_i^+}$. Further, let*

$$u_i^* := \arg\max_{u_i \in \overline{\mathcal{U}}_i} \mathcal{L}_{g_i} h_i(x_i) u_i \tag{21}$$

and

$$\begin{aligned} u_i^c(\nu_i) := \arg\max_{u_i \in \overline{\mathcal{U}}_i} \quad & u_i^\top Q_i(x_i) u_i + \mathcal{L}_{g_i} h_i(x_i) \tilde{d}(u_i) \\ & + \left[q_i(\mathbf{x}_i) + \nu_i \mathcal{L}_{g_i} h_i(x_i)\right] u_i, \end{aligned} \tag{22}$$

where the optimizer $u_i^c(\nu_i)$ is a continuous function of ν_i. Then, we have that $\lim_{\nu_i \to \infty} u_i^c(\nu_i) = u_i^*$.

Proof. Note that in (22) as we increase ν_i the linear term $\mathcal{L}_{g_i} h_i(x_i)$ will begin to dominate the expression for the capability of node $i \in [n]$. Therefore, by Assumptions 2 and 3 we have

$$\begin{aligned} \lim_{\nu_i \to \infty} \frac{1}{\nu_i} \Big(u_i^\top Q_i(x_i) u_i + q_i(\mathbf{x}_i) u_i + \mathcal{L}_{g_i} h_i(x_i) \tilde{d}(u_i) \Big) + \mathcal{L}_{g_i} h_i(x_i) u_i \\ = \mathcal{L}_{g_i} h_i(x_i) u_i. \end{aligned}$$

Thus, since $u_i^c(\nu_i)$ is a continuous function of ν_i, we have, by [26, Theorem 4.1], that $\lim_{\nu_i \to \infty} u_i^c(\nu_i) = u_i^*$, where u_i^* is unique by Lemma 1. □

Given these results, we may quantify the discrepancy between the communicated capability and the actual capability of node i with respect to a chosen ν_i. We define the quantity of neighbor compliance as follows. First, define the minimum assistance requested from neighbor $j \in \mathcal{N}_i^+$ by node i as the set of control actions

$$U_j^m(x) = \{u_j \in \overline{\mathcal{U}}_j : a_{ij}(\mathbf{x}_i) u_j + \bar{c}_{ij} = 0\}, \tag{23}$$

where $\bar{c}_{ij}$ is the safety commitment allocated to node j by node i according to Algorithm 1. We compute the amount of *compliance* by neighbor j to requests by node i given any action $u_j \in \mathcal{U}_j$ as

$$e_{ij}(\mathbf{x}_i, u_j) := a_{ij}(\mathbf{x}_i)(u_j^m - u_j), \tag{24}$$

where we may choose any $u_j^m \in U_j^m(\mathbf{x}_i)$ without loss of generality.

Definition 2. *If $e_{ij}(\mathbf{x}_i, u_j) < 0$, then neighbor j is* ***non-compliant****. The* ***neighborhood compliance*** *of incoming neighbors to node i is given by $e_i(\mathbf{x}_i, u_{\mathcal{N}_i^+}) = \sum_{j \in \mathcal{N}_i^+} e_{ij}(\mathbf{x}_i, u_j)$, where if $e_i(\mathbf{x}_i, u_{\mathcal{N}_i^+}) < 0$, then the incoming neighborhood of node i is* ***non-compliant****.*

Note that, in our definition of non-compliance, we consider any action u_j for neighbor $j \in \mathcal{N}_i^+$ to be non-compliant if it violates the safety requirement of node i. We make the following assumption.

Assumption 5. *There exists $\nu_i^* < \infty$ such that $u_i^c(\nu_i^*) = u_i^*$.*

We may then compute the upper bound on the total non-compliance agent i is resilient to as follows.

Theorem 1. *For a given $\mathbf{x}_i \in \mathbb{R}^{N_i + N_i^+}$, let $\nu_i \leq \nu_i^*$, under Assumption 5, with u_i^* and u_i^c be computed via* (21) *and* (22)*, respectively. If Assumptions 1-4 hold and $e_i(\mathbf{x}_i) < 0$, then node i will be capable of satisfying its safety constraints in the presence of non-compliance from its incoming neighborhood up to*

$$\begin{aligned} |e_i(\mathbf{x}_i, u_{\mathcal{N}_i^+})| \leq{} & (u_i^* - u_i^c)^\top Q_i(x_i)(u_i^* - u_i^c) + q_i(\mathbf{x}_i)(u_i^* - u_i^c) \\ & + \mathcal{L}_{g_i} h_i(x_i)(\tilde{d}(u_i^*) - \tilde{d}(u_i^c)) \\ & + \nu_i^* \mathcal{L}_{g_i} h_i(x_i) u_i^* - \nu_i \mathcal{L}_{g_i} h_i(x_i) u_i^c. \end{aligned} \tag{25}$$

Proof. By Assumption 4 and Lemma 1 we that $u_i^* \in \partial\overline{\mathcal{U}}_i$ is unique and therefore $\mathcal{L}_{g_i} h_i(x_i) u_i^* > \mathcal{L}_{g_i} h_i(x_i) u_i, \forall u_i \in \overline{\mathcal{U}}_i \setminus \{u_i^*\}$. Thus, for all $\nu_i \leq \nu_i^*$, we have

$$c_i(\mathbf{x}_i^+, u_i^*, \nu_i^*) \geq c_i(\mathbf{x}_i^+, u_i^c, \nu_i) \tag{26}$$

for a given $x \in \mathbb{R}^N$. Consider when

$$c_i(\mathbf{x}_i^+, u_i^*, \nu_i^*) \leq 0$$

and there exists $\nu_i > 0$ such that

$$\psi_i^2(\mathbf{x}_i^+, u_i^c, u_{\mathcal{N}_i^+}^c, \nu_i) = \psi_i^2(\mathbf{x}_i^+, u_i^*, u_{\mathcal{N}_i^+}^*, \nu_i^*). \tag{27}$$

Thus, by (27), we have

$$\sum_{j \in \mathcal{N}_i^+} a_{ij}(u_j^* - u_j^c) + c_i(\mathbf{x}_i^+, u_i^*, \nu_i^*) - c_i(\mathbf{x}_i^+, u_i^c, \nu_i) = 0$$

and by (26) we have

$$\sum_{j \in \mathcal{N}_i^+} a_{ij}(u_j^* - u_j^c) \leq 0.$$

In this way, we characterize the difference between the true maximum capability of node i and its communicated maximum capability as

$$\begin{aligned} \epsilon_i(\mathbf{x}_i, \nu_i) :={} & (u_i^* - u_i^c)^\top Q_i(x_i)(u_i^* - u_i^c) + q_i(\mathbf{x}_i)(u_i^* - u_i^c) \\ & + \mathcal{L}_{g_i} h_i(x_i)(\tilde{d}(u_i^*) - \tilde{d}(u_i^c)) \\ & + \nu_i^* \mathcal{L}_{g_i} h_i(x_i) u_i^* - \nu_i \mathcal{L}_{g_i} h_i(x_i) u_i^c \geq 0, \end{aligned} \tag{28}$$

where, by Assumptions 2-4 and Lemma 2, we have if $\nu_i \geq \nu_i^*$, then $\epsilon_i(\mathbf{x}_i, \nu_i) = 0$. Therefore, if we consider any deficiency between the minimum requested assistance via Algorithm 1 and the actual actions of neighbors $e_{ij} = a_{ij}(u_j^m - u_j)$ for $j \in \mathcal{N}_i^+$, if $\left|\sum_{j\in\mathcal{N}_i^+} e_{ij}\right| \leq \epsilon_i(\mathbf{x}_i, \nu_i)$ then there exists a $u_i^* \in \overline{\mathcal{U}}_i$ and ν_i^* such that $\sum_{j\in\mathcal{N}_i^+} a_{ij}(\mathbf{x}_i)u_j + c_i(\mathbf{x}_i^+, u_i^*, \nu_i^*) \geq 0$. □

Note that u_i^* is a function of x_i and u_i^c is a function of $\mathbf{x}_i$ and ν_i, making (25) an implicitly time-varying function. Thus, by Theorem 1 we may compute the resilience that node i has to neighborhood non-compliance with respect to a given neighborhood state and choice of $\nu_i = \eta_i + \kappa_i$. Further, note that if $\nu_i \geq \nu_i^*$, then $u_i^c = u_i^*$ and (25) becomes $|e_i| = 0$. Thus, we can use ν_i^* to characterize a kind of *resilience boundary* for a given 1-hop neighborhood state $\mathbf{x}_i$ at a given time, where any $\nu_i \geq \nu_i^*$ characterizes a system with no resilience. Note that, given the time-varying nature of (25), a given ν_i^* that induces resilience for node i at one state $\mathbf{x}_i$ may not induce resilience for another state, meaning that ν_i^* is also time-varying (see Figure 3).

This measure of resilience may be useful in considering parameter selection for each node in the network, depending on the reliability of neighboring nodes and the consequences of node failure. However, it should also be noted that, as (28) increases, node i will place a greater burden of its safety on its neighbors $j \in \mathcal{N}_i^+$, which could lead to an over-taxing of neighbor control resources and under-utilization of the resources of node i. Therefore, the context and consequences of over-requesting by neighbors in specific applications should be considered when selecting η_i and κ_i. We illustrate these behaviors on networked epidemic processes in the following section.

V. Simulations

We define an SIS networked epidemic spreading process [5] with n nodes where x_i is the proportion of infected population at node $i \in [n]$, with the infection dynamics defined by

$$\dot{x}_i = -(\gamma_i + u_i)x_i + (1 - x_i)\sum_{j\in\mathcal{N}_i^+} \beta_{ij}x_j, \tag{29}$$

where $\gamma_i > 0$ is the recovery rate at node i, $u_i \in \mathcal{U}_i \subset \mathbb{R}_{\geq 0}$ is a control input boosting the healing rate at node i, and $\beta_{ij} \geq 0$ is the networked connection going from node j to node i (note β_{ii} is simply the infection rate occurring at node i).

Let $\mathcal{U}_i = [0, \bar{u}_i]$, where $\bar{u}_i$ is the upper limit on boosting the healing rate at node i. We let each node define its individual safety constraint as

$$h_i(x_i) = \bar{x}_i - x_i, \tag{30}$$

where $\bar{x}_i \in (0, 1]$ is the defined safety threshold for the acceptable proportion of infected individuals at node i. Thus, the individual safe sets for each node are given by

$$\mathcal{C}_i = \{x_i \in [0, 1] : h_i(x_i) \geq 0\}. \tag{31}$$

For this example, we construct a simple 3-node system where $\beta_{ij} = 0.25$ and $\beta_{ii} = 0.5$, for all $j \neq i$ and $j = i$, respectively. To induce an endemic state, we set the healing rate $\gamma_i = 0.3$, for all nodes. We set the safety constraints for each node as $\bar{x}_1 = 0.1$, $\bar{x}_2 = 0.12$, and $\bar{x}_3 = 0.18$ and set the initial infection levels at $x = [0.04, 0.01, 0.02]$.

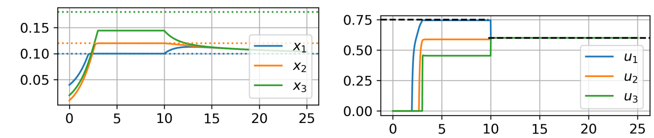

Fig. 1: Simulated networked SIS model from (29), with each node implementing the collaborative safety framework from [20], where $n = 3$ and $\eta_i = 10, \kappa_i = 1$ for all $i \in [3]$. Each node's safety constraint $\bar{x}_i$ is represented by the dotted line of corresponding color and the input constraint $\mathcal{U}_i(t)$ for all nodes $i \in [3]$ is shown by the black dotted line, where $\mathcal{U}_i(t) = [0, 0.75]$ if $t < 10$ and $\mathcal{U}_i(t) = [0, 0.6]$ if $t \geq 10$. Note that the system failure at $t = 10$ causes node 1 to violate its safety constraint.

To illustrate how $\nu_i = \eta_i + \kappa_i$ can be used to increase resilience to sudden changes or system failures, we simulate the same system for both large ν_i ($\eta_i = 10$ and $\kappa_i = 1$) and small ν_i ($\eta_i = 0.3$ and $\kappa_i = 0.3$), for all $i \in [3]$, in Figures 1 and 2, respectively. In both simulations, we set the control limits $\mathcal{U}_i(t) = [0, 0.75]$ for $t < 10$ and then perturb the system by suddenly reducing the control ability to $\mathcal{U}_i(t) = [0, 0.6]$ for $t \geq 10$, for all $i \in [3]$. Note that the collaborative safety framework maintains safety for both instances up to $t = 10$, and when the system experiences the failure at $t = 10$ the dynamics react accordingly. For the less resilient system, shown in Figure 1, we see node 1 violates its safety requirement after the system failure at $t = 10$. By contrast, we see in Figure 2 that when ν_i is small for all $i \in [3]$, it causes node 1 to over-request assistance from its neighbors, leaving increased control ability for node 1 that allows it to be resilient to the network failure, consistent with the result in Theorem 1. Thus we see that the network is able to operate reliably even after experiencing some failures when ν_i is sufficiently small for all i, i.e., when $\nu_i \ll \nu_i^*$.

Recall from Assumption 5 that ν_i^* is the minimum ν_i such that $u_i^c = u_i^*$, where u_i^* and u_i^c are computed via (21) and (22), respectively. To visualize the resilience boundary, ν_i^*, of node 1 for a given network state, we fix $x_2 = x_3 = 0.1$ and approximate ν_1^* in Figure 3 by incrementing ν_1 by $\delta_\nu = 0.01$ until $u_1^* = u_1^c$ for $x_1 = [0, \bar{x}_1]$. In Figure 3, we see that, as x_1 approaches $\bar{x}_1$, the resilience boundary increases to include larger values of ν_1 that would enable resilience for node 1 to non-compliant neighbors. We then use this approximation of ν_i^* to generate the surface in Figure 4 by computing $\epsilon_i(x, \nu_i)$ using (28) for $x_i \in [0, \bar{x}_1]$ and $\nu_1 \in [0, 0.8]$. Note that for small x_1 we have that the linear term $q_1(x) = \mathcal{L}_{g_i}\mathcal{L}_{f_i}h_i(\mathbf{x}_i)^\top + \mathcal{L}_{g_i}\mathcal{L}_{f_i}h_i(\mathbf{x}_i)$ overpowers the negative definite quadratic term $Q_1(x_1)$, thereby setting $\nu_1^* = 0$. Therefore, we see that the tolerance for neighborhood non-compliance grows as ν_1 decreases, and as x_1 approaches the barrier of its safety constraints it will require a larger amount of error tolerance from its neighbors.

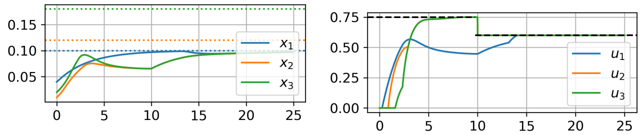

Fig. 2: The same simulation from Figure 1 except $\eta_i = \kappa_i = 0.3$ for all $i \in [3]$. Note that, unlike in Figure 1, node 1 is able to satisfy its safety constraint for all t.

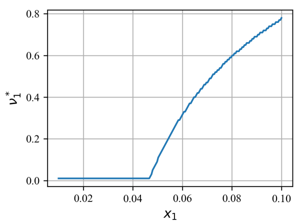

Fig. 3: A numerical approximation of ν_1^* for $x_2 = x_3 = 0.1$ with $x_1 \in [0, \bar{x}_1]$, where ν_1 is incremented by $\delta_\nu = 0.01$ until $u_1^* = u_1^c$. Notice as x_1 approaches $\bar{x}_1$, the resilience boundary increases to include larger values of ν_1 that would enable resilience for node 1 to non-compliant neighbors.

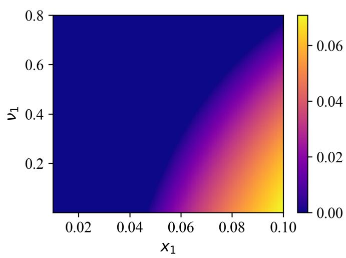

Fig. 4: The error tolerance in neighbor request fulfillment $\epsilon_i(\mathbf{x}_i, \nu_i)$, defined by (28), computed for $x_2 = x_3 = 0.1$ with $x_1 \in [0, \bar{x}_1]$ and $\nu_1 \in [0, 0.8]$. Notice as x_1 approaches the barrier of its safety constraints it requires a larger amount of error tolerance from its neighbors.

VI. Conclusion

In this letter, we have explored properties of resilience for coupled agents with respect to non-compliance to safety requests made to neighbors under a collaborative safety framework. We have proposed an upper bound on the total non-compliance each agent may tolerate while still being able to satisfy their individual safety needs, given its 1-hop neighborhood states and knowledge of networked dynamics. Future work includes computing the time-varying upper bound ν_i^* explicitly, incorporating this bound directly into the implementation of Algorithm 1, and extending this property of non-compliance to include general uncertainty and noise in the coupled dynamic system. Additionally, thus far our collaborative framework assumes all agents are cooperative; however, since the notion of non-compliance naturally implies scenarios where some agents may act maliciously, further work will account for such non-cooperative agents.

References

[1] M. L. Tuballa and M. L. Abundo, "A review of the development of smart grid technologies," *Renewable and Sustainable Energy Reviews*, vol. 59, pp. 710–725, 2016.

[2] A. Tahir, J. Böling, M.-H. Haghbayan, H. T. Toivonen, and J. Plosila, "Swarms of unmanned aerial vehicles—A survey," *Journal of Industrial Information Integration*, vol. 16, p. 100106, 2019.

[3] S.-J. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar, "A survey on aerial swarm robotics," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 837–855, 2018.

[4] L. Wang, A. D. Ames, and M. Egerstedt, "Safety barrier certificates for collisions-free multirobot systems," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 661–674, 2017.

[5] P. E. Paré, C. L. Beck, and T. Başar, "Modeling, estimation, and analysis of epidemics over networks: An overview," *Annual Reviews in Control*, vol. 50, pp. 345–360, 2020.

[6] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, no. 11, pp. 1747–1767, 1999.

[7] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2016.

[8] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *Proc. European Control Conference (ECC)*, 2019, pp. 3420–3431.

[9] F. L. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative Control of Multi-agent Systems: Optimal and Adaptive Design Approaches*. Springer Science & Business Media, 2013.

[10] L. Lindemann and D. V. Dimarogonas, "Control barrier functions for multi-agent systems under conflicting local signal temporal logic tasks," *IEEE Control Systems Letters*, vol. 3, no. 3, pp. 757–762, 2019.

[11] X. Tan and D. V. Dimarogonas, "Distributed implementation of control barrier functions for multi-agent systems," *IEEE Control Systems Letters*, vol. 6, pp. 1879–1884, 2021.

[12] N. De Carli, P. Salaris, and P. R. Giordano, "Distributed control barrier functions for global connectivity maintenance," in *Proc. IEEE International Conf on Robotics and Automation*, 2024, pp. 1–7.

[13] P. Mestres, C. Nieto-Granda, and J. Cortés, "Distributed safe navigation of multi-agent systems using control barrier function-based controllers," *IEEE Robotics and Automation Letters*, 2024.

[14] H. Parwana, A. Mustafa, and D. Panagou, "Trust-based rate-tunable control barrier functions for non-cooperative multi-agent systems," in *Proceedings of the IEEE Conference on Decision and Control*. IEEE, 2022, pp. 2222–2229.

[15] R. K. Ramachandran, P. Pierpaoli, M. Egerstedt, and G. S. Sukhatme, "Resilient monitoring in heterogeneous multi-robot systems through network reconfiguration," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 126–138, 2021.

[16] S. Dobson, D. Hutchison, A. Mauthe, A. Schaeffer-Filho, P. Smith, and J. P. Sterbenz, "Self-organization and resilience for networked systems: Design principles and open research issues," *Proceedings of the IEEE*, vol. 107, no. 4, pp. 819–834, 2019.

[17] R. Filippini and A. Silva, "A modeling framework for the resilience analysis of networked systems-of-systems based on functional dependencies," *Reliability Engineering & System Safety*, vol. 125, pp. 82–91, 2014.

[18] D. A. Reed, K. C. Kapur, and R. D. Christie, "Methodology for assessing the resilience of networked infrastructure," *IEEE Systems Journal*, vol. 3, no. 2, pp. 174–180, 2009.

[19] N. Criado, E. Argente, and V. Botti, "Open issues for normative multi-agent systems," *AI Communications*, vol. 24, no. 3, pp. 233–264, 2011.

[20] B. A. Butler and P. E. Paré, "Collaborative safety-critical control for dynamically coupled networked systems," 2024. [Online]. Available: https://arxiv.org/abs/2310.03289

[21] B. A. Butler, C. H. Leung, and P. E. Paré, "Collaborative safe formation control for coupled multi-agent systems," in *Proc. European Control Conference (ECC)*. IEEE, 2024, pp. 3410–3415.

[22] L. D. Valdez, L. Shekhtman, C. E. La Rocca, X. Zhang, S. V. Buldyrev, P. A. Trunfio, L. A. Braunstein, and S. Havlin, "Cascading failures in complex networks," *Journal of Complex Networks*, vol. 8, no. 2, p. cnaa013, 2020.

[23] K. Röbenack, "Computation of multiple Lie derivatives by algorithmic differentiation," *Journal of Computational and Applied Mathematics*, vol. 213, no. 2, pp. 454–464, 2008.

[24] W. Xiao and C. Belta, "High-order control barrier functions," *IEEE Transactions on Automatic Control*, vol. 67, no. 7, pp. 3655–3662, 2021.

[25] X. Tan, W. S. Cortez, and D. V. Dimarogonas, "High-order barrier functions: Robustness, safety, and performance-critical control," *IEEE Transactions on Automatic Control*, vol. 67, no. 6, pp. 3021–3028, 2021.

[26] G. Still, *Lectures on Parametric Optimization: An Introduction*. Optimization Online, 2018.