Decentralized Monitoring of Leader-Follower Networks of Uncertain Nonlinear Systems

J. R. Klotz, L. Andrews, R. L. Kamalapurkar, and W. E. Dixon

***Abstract*—Efforts in this paper seek to develop a new method to monitor for undesirable performance in the general leader-follower network structure of autonomous agents. Concepts from optimal control and adaptive dynamic programming (ADP) are used to develop a novel metric which networked agents with uncertain nonlinear dynamics use to monitor each other with decentralized communication. The developed approach uses a data-driven concurrent learning-based policy to identify agent dynamics and functions used to characterize optimality conditions, which are then used to check for compliance with specified performance criteria.**

I. Introduction

Implementing a network of cooperating agents (e.g., flocks of UAVs, teams of ground vehicles) helps to ensure mission completion and provides more advanced tactical capabilities. Networks containing agents enacting decentralized control policies, wherein only information from neighboring agents is used to internally make decisions, benefit from autonomy: each agent is encoded with a (possibly disaggregated) tactical mission objective and has no need to maintain contact with a mission coordinator.

However, networked systems must be cognizant of the reliability of neighbors' influence, especially if the systems are autonomous; the shift of behavioral responsibility to a stand-alone computer requires precautions. For example, network neighbors may be unreliable due to input disturbances, faulty dynamics, or network subterfuge, such as cyber-attacks. Cyber-attacks are genuine threats to networked systems and require strong preventative measures, evidenced by the Pentagon's plan of an expansion of its cybersecurity force to reduce the nation's vulnerability to hackers who could "dismantle the nation's power grid" and interfere with other critical infrastructure systems [1]. In fact, cyber-sabotage of networked systems has already taken place: supervisory control and data acquisition (SCADA) systems, used prevalently to monitor and control power networks, water distribution systems, and oil and gas pipelines, have been maliciously compromised, resulting in much damage to large networks [2], [3].

J. R. Klotz, L. Andrews, R. Kamalapurkar, and W. E. Dixon are with the Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville FL 32611-6250 USA. Email: {jklotz, landr010, rkamalapurkar, wdixon}@ufl.edu.

This research is supported in part by NSF award numbers 1161260, 1217908, and the AFRL Mathematical Modeling and Optimization Institute. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the sponsoring agency.

Efforts have been made to structure the layout of a network such that the impact of network corruption can be abated [4], [5]. Researchers have also investigated the ability to allay types of network subterfuge by creating control algorithms which are resilient to attacks on sensors and actuators [6]. Other efforts seek to have agents detect undesired performance in their network neighbors. The results in [7] and [8] provide methods to detect "sudden" faulty behavior which is modeled as a step function multiplied by fault dynamics. Other works develop procedures to detect generally undesired behavior in networks of linear systems [9], [10] and unpredicted state trajectories of nonlinear systems using neural networks (NN) [11], [12]. Adaptive thresholds used for determining if the state of a neighboring agent is within an acceptable tolerance are developed in [13] and [14].

An issue with only using a neighbor's state to judge performance is that, depending on the dynamics, a small deviation in the control may cause a large deviation away from the expected state of the system at the following time step; the converse may also occur. Thus, if judging only by the trajectory of a dynamical system, minimally deviant behavior may be exaggerated during monitoring or significantly deviant behavior may not be noticed. Thus, motivation exists to examine more information than just the state when judging an agent's behavior. The intuition behind considering both state errors and control effort is clear upon recalling that both state errors and control effort are used in common cost functions, such as that in LQR control.

The contribution of this paper is the development of a novel metric, based on the Bellman error (BE), which provides a condition for determining if a network neighbor with uncertain nonlinear dynamics is behaving near optimally; furthermore, this monitoring procedure only requires neighbor communication and may be implemented online. The contribution is facilitated by the use of adaptive dynamic programming (ADP) and concurrent learning to approximately determine how close optimality conditions are to being satisfied.

II. Problem Description

A. Graph theory preliminaries

Consider a network of N agents with a communication topology described by the directed graph $\mathcal{G} = \{\mathcal{V}, E\}$, where $\mathcal{V} = \{1, 2, \ldots, N\}$ is the set of agents and $E \subseteq \mathcal{V} \times \mathcal{V}$ are the corresponding communication links. The set E contains an ordered pair (j, i) such that $(j, i) \in E$ if agent j communicates information to agent i. The neighborhood of

agent i is defined as $\mathcal{N} \triangleq \{j \in \mathcal{V} \mid (j,i) \in E\}$, the set of all agents which communicate to i. It is assumed that the graph is simple, i.e., there are no self loops: $(i,i) \notin E$. Each communication link is weighted by a constant $a_{ij} \in \mathbb{R}$, where $a_{ij} > 0$ if $(j,i) \in E$ and $a_{ij} = 0$ otherwise. The graph adjacency matrix $A \in \mathbb{R}^{N \times N}$ is constructed from these weights as $A \triangleq [a_{ij} \mid i,j \in \mathcal{V}]$.

The interaction of the network leader with the other agents is described by the graph $\bar{\mathcal{G}} = \left\{\bar{\mathcal{V}}, \bar{E}\right\}$, which is a supergraph of $\mathcal{G}$ that includes the leader, denoted by 0, such that $\bar{\mathcal{V}} = \mathcal{V} \cup \{0\}$. The communication link set $\bar{E}$ is constructed such that $\bar{E} \supset E$ and $(0,i) \in \bar{E}$ if the leader communicates to i. The leader-included neighborhood is accordingly defined as $\bar{\mathcal{N}}_i \triangleq \left\{j \in \bar{\mathcal{V}} \mid (j,i) \in \bar{E}\right\}$. Leader connections are weighted by the pinning matrix $A_0 \in \mathbb{R}^{N \times N}$, where $A_0 \triangleq \operatorname{diag}(a_{i0}) \mid i \in \mathcal{V}$ and $a_{i0} > 0$ if $(0,i) \in \bar{E}$ and $a_{i0} = 0$ otherwise.

B. Problem definition

Consider a leader-follower network in which the follower agents synchronize in state towards the leader's (possibly time-varying) state. Let the possibly heterogeneous dynamics for agent $i \in \mathcal{V}$ be described in general first-order form as

$$\dot{x}_i = f_i(x_i) + g_i u_i, \tag{1}$$

where $x_i \in S$ is the state, the set $S \subset \mathbb{R}^n$ is the space in which the agents' states lie, $f_i : S \to \mathbb{R}^n$ is a locally Lipschitz function, $g_i \in \mathbb{R}^{n \times m}$ is a known constant matrix, and $u_i \in \mathbb{R}^m$ is a pre-established, stabilizing, synchronizing control input.

The monitoring objective applied at each agent is to cooperatively monitor the network for satisfaction of its control objective, wherein the network may be affected by input disturbances that cause suboptimal performance. Moreover, the monitoring protocol should be decentralized and passive, i.e., only information from one-hop neighbors should be used and the protocol should not interfere with the monitored systems.

C. Approach

For typical synchronization techniques, such as model predictive control (MPC), inverse-optimal, or adaptive dynamic programming (ADP), a control law is developed based on a cost function of the form

$$J_i \triangleq \int_0^{t_f} \left(Q_i(e_i) + u_i^T R_i u_i\right) d\tau, \tag{2}$$

where $t_f > 0$ is the final time of the optimal control problem, $Q_i : \mathbb{R}^n \to \mathbb{R}$ is a tracking error weighting function, R_i is a constant positive definite symmetric weighting matrix, and e_i is the neighborhood tracking error defined as

$$e_i \triangleq \sum_{j \in \mathcal{N}_i} a_{ij}(x_i - x_j) + a_{i0}(x_i - x_0).$$

Even if a controller is not developed based on a cost function, such as in robust and adaptive control, techniques exist which can be used to develop an expression for a meaningful cost function in the form of (2) for a given control policy (cf. [15]–[17]). The following monitoring approach uses the cost function in (2) to observe how well the networked dynamical systems are satisfying optimality conditions; specifically, satisfaction of the Hamilton-Jacobi-Bellman (HJB) equation will be monitored to determine how "closely" to optimal the networked systems are operating. Before the monitoring protocol is given, some optimal control concepts are introduced. Based on Bellman's principle of optimality, an equivalent representation of the optimal control problem in (2) is given by the value function $V_i : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$, defined as $V_i(e_i,t) \triangleq \int_t^{t_f} \left(Q_i(e_i) + u_i^T R_i u_i\right) d\tau$, which is minimized as

$$V_i^* = \min_{u_i \in \mathbb{U}_i} \int_t^{t_f} \left(Q_i(e_i) + u_i^T R_i u_i\right) d\tau,$$

where $\mathbb{U}_i$ is the set of admissible feedback control policies for agent i [18] and the superscript $(\cdot)^*$ denotes minimizing (optimal) conditions. Because the minimization of the value function V_i is inherently coupled with the minimization of other value functions in the network, the value function V_i can naturally be a function of the error signal e_j, $j \in \mathcal{V}$, if there exists a directed path from the leader to agent i that includes agent j. The resulting coupled Hamiltonian for agent i is defined by the function

$$\mathcal{H}_i(\mathcal{E},\mathcal{X},\mathcal{U},V_i,t) \triangleq Q_i(e_i) + u_i^T R_i u_i + \sum_{j \in \mathcal{V}} \frac{\partial V_i(\mathcal{E})}{\partial e_j} \times \sum_{k \in \bar{\mathcal{N}}_j} a_{jk}\left(f_j(x_j) + g_j u_j - f_k(x_k) - g_k u_k\right), \tag{3}$$

where the sets $\mathcal{E}$, X and U are defined as $\mathcal{E} \triangleq \{e_i \mid i \in \mathcal{V}\}$, $\mathcal{X} \triangleq \{x_i \mid i \in \mathcal{V}\}$, and $\mathcal{U} \triangleq \{u_i \mid i \in \mathcal{V}\}$. Assuming that $V_i^* \in \mathcal{C}^1$, V_i^* and the Hamiltonian H_i are used to construct the coupled HJB optimality constraint as [19]

$$\frac{\partial V_i^*}{\partial t} + \mathcal{H}_i(\mathcal{E},\mathcal{X},\mathcal{U}^*,V_i^*,t) \equiv 0. \tag{4}$$

Thus, one method for monitoring the network's operating conditions is to monitor the expression $\frac{\partial V_i^*}{\partial t} + \mathcal{H}_i(\mathcal{E},\mathcal{X},\mathcal{U},V_i^*,t)$, which equals zero for the implementation of optimal control efforts.

Because the optimal value function V_i^* is often infeasible to solve analytically, an ADP-based approximation scheme is subsequently developed so that the approximate value of $\frac{\partial V_i^*}{\partial t} + \mathcal{H}_i(\mathcal{E},\mathcal{X},\mathcal{U},V_i^*,t)$ may be monitored. However, as seen in (3), the HJB constraint for agent i in (4) is inherently coupled with the state and control of every agent $j \in \mathcal{V}$ such that there exists a directed path from the leader to agent i. Consequently, checking for satisfaction of the HJB is seemingly unavoidably centralized in information communication. To overcome this restriction, the developed ADP-based approximation scheme is constructed such that only information concerning one-hop neighbors' states, neighbors' control policies and time is necessary for value function approximation.

The use of ADP to monitor a dynamical system imposes a serious challenge: persistence of excitation (PE) is typically used to guarantee function approximation. This is typically achieved by injecting a frequency-rich dither signal into the control input of the dynamical system to provide enough volatility, or data richness, in the regressor vector used by the ADP protocol. However, doing so for this application would interfere with the performance of the monitored systems. To overcome this obstacle, a concurrent learning-based technique is used in the developed ADP based protocol, which obtains data richness without PE, thus providing the ability to create a "passive", or non-interfering, monitoring scheme.

To make the current problem tenable, it is assumed that authentic information is exchanged between the agents, i.e., communication is not maliciously compromised; rather, the agents are cooperatively monitoring each other's performance. If necessary, communication authentication algorithms such as in [9] or [20] can be used to verify digitally communicated information.

To evaluate the expression in (3), knowledge of the drift dynamics f_i is required. The following section provides a method to estimate the function f_i using a data-based approach. Some function arguments are omitted in the remainder of the paper for the sake of brevity where the meaning is clear.

III. System Identification

Assumption 1. The uncertain, locally Lipschitz drift dynamics, f_i, are linear-in-the-parameters (LP), such that $f_i(x_i) = Y_i(x_i)\theta_i^*$, where $Y_i : \mathbb{R}^n \to \mathbb{R}^{n\times p_i}$ is a known regression matrix, and $\theta_i^* \in \mathbb{R}^{p_i}$ is a vector of constant unknown parameters.

The function $\hat{f}_i : \mathbb{R}^n \times \mathbb{R}^{p_i} \to \mathbb{R}^n$ is an estimate of the uncertain drift dynamics f_i and is defined as $\hat{f}_i\left(x_i, \hat{\theta}_i\right) \triangleq Y_i(x_i)\hat{\theta}_i$, where $\hat{\theta}_i \in \mathbb{R}^{p_i}$ is an estimate of the unknown vector θ_i^*. The estimation of θ_i^* is facilitated by the construction of the identifier

$$\dot{\hat{x}}_i = \hat{f}_i + g_i u_i + k_{xi}\tilde{x}_i, \tag{5}$$

where $\tilde{x}_i \triangleq x_i - \hat{x}_i$ is the state estimation error, and $k_{xi} \in \mathbb{R}^{n\times n}$ is a constant positive definite diagonal gain matrix. The state identification error dynamics are expressed using (1) and (5) as

$$\dot{\tilde{x}}_i = Y_i\tilde{\theta}_i - k_{xi}\tilde{x}_i, \tag{6}$$

where $\tilde{\theta}_i \triangleq \theta^* - \hat{\theta}$. The state estimator in (5) is used to develop a data-driven concurrent learning-based update law for $\hat{\theta}_i$ as

$$\dot{\hat{\theta}}_i = \Gamma_{\theta i}(Y_i)^T\tilde{x}_i + \Gamma_{\theta i}k_{\theta i}\sum_{\xi=1}^{K}\left(Y_i^\xi\right)^T\left(\dot{x}_i^\xi - g_i u_i^\xi - Y_i^\xi\hat{\theta}_i\right), \tag{7}$$

where $\Gamma_{\theta i} \in \mathbb{R}^{p_i\times p_i}$ is a constant positive definite symmetric gain matrix, $k_{\theta i} \in \mathbb{R}_{>0}$ is a constant concurrent learning gain, and the superscript $(\cdot)^\xi$ denotes evaluation at one of the unique recorded values in the state data stack $\left\{x_i^\xi \mid \xi = 1,\ldots,K\right\}$ or corresponding control value data stack $\left\{u_i^\xi \mid \xi = 1,\ldots,K\right\}$. It is assumed that these data stacks are recorded prior to use of the drift dynamics estimator. The following assumption specifies the necessary data richness of the recorded data.

Assumption 2. [21] There exists a finite set of collected data $\left\{x_i^\xi \mid \xi = 1,\ldots,K\right\}$ such that

$$\operatorname{rank}\left(\sum_{\xi=1}^{K}\left(Y_i^\xi\right)^T Y_i^\xi\right) = p_i. \tag{8}$$

Note that PE is not mentioned as a necessity for this identification algorithm; instead of guaranteeing data richness by assuming that the dynamics are persistently exciting for all time, it is only assumed that there exists a *finite* set of data points that provide the necessary data richness. This also eliminates the common requirement for injection of a persistent dither signal to attempt to ensure PE, which would interfere with the monitored systems. Furthermore, contrary to PE-based approaches, the condition in (8) can be verified. Note that, because (7) depends on the state derivative $\dot{x}_i^\xi$ at a past value, numerical techniques can be used to approximate $\dot{x}_i^\xi$ using preceding and proceeding recorded state information.

To facilitate an analysis of the performance of the identifier in (7), the identifier dynamics are expressed in terms of estimation errors as

$$\dot{\tilde{\theta}}_i = -\Gamma_{\theta i}(Y_i)^T\tilde{x}_i - \Gamma_{\theta i}k_{\theta i}\sum_{\xi=1}^{K}\left(Y_i^\xi\right)^T\left(\dot{x}_i^\xi - g_i u_i^\xi - Y_i^\xi\hat{\theta}_i\right). \tag{9}$$

To describe the performance of the identification of θ_i^*, consider the positive definite continuously differentiable Lyapunov function $V_{\theta i} : \mathbb{R}^{n+p_i} \to [0,\infty)$ defined as

$$V_{\theta i}(z_i) \triangleq \frac{1}{2}\tilde{x}_i^T\tilde{x}_i + \frac{1}{2}\tilde{\theta}_i^T\Gamma_{\theta i}^{-1}\tilde{\theta}_i, \tag{10}$$

where $z_i \triangleq \begin{bmatrix}\tilde{x}_i^T, & \tilde{\theta}_i^T\end{bmatrix}^T$.The expression in (10) satisfies the inequalities

$$\underline{V}_{\theta i}\left\|z_i\right\|^2 \le V_{\theta i}(z_i) \le \bar{V}_{\theta i}\left\|z_i\right\|^2, \tag{11}$$

where $\underline{V}_{\theta i} \triangleq \frac{1}{2}\min\left(1, \lambda_{\min}\left(\Gamma_{\theta i}^{-1}\right)\right)$ and $\bar{V}_{\theta i} \triangleq \frac{1}{2}\max\left(1, \lambda_{\max}\left(\Gamma_{\theta i}^{-1}\right)\right)$ are positive known constants and the operations $\lambda_{\min}(\cdot)$ and $\lambda_{\max}(\cdot)$ denote the minimum and maximum eigenvalues, respectively. Using the dynamics in (6) and (9), the time derivative of (10) is expressed as

$$\dot{V}_{\theta i} = -\tilde{x}_i^T k_{xi}\tilde{x}_i - \tilde{\theta}_i^T k_{\theta i}\left(\sum_{\xi=1}^{K}\left(Y_i^\xi\right)^T Y_i^\xi\right)\tilde{\theta}_i. \tag{12}$$

Note that because the matrix $\sum_{\xi=1}^{K}\left(Y_i^\xi\right)^T Y_i^\xi$ is symmetric and positive semi-definite, its eigenvalues are real and

greater than or equal to zero. Furthermore, by Assumption 2, none of the eigenvalues of $\sum_{\xi=1}^{K}\left(Y_i^{\xi}\right)^T Y_i^{\xi}$ are equal to zero. Thus, all of the eigenvalues of the symmetric matrix $\sum_{\xi=1}^{K}\left(Y_i^{\xi}\right)^T Y_i^{\xi}$ are positive, and the matrix $\sum_{\xi=1}^{K}\left(Y_i^{\xi}\right)^T Y_i^{\xi}$ is positive definite. Using this property and the inequalities in (11), (12) is upper bounded as

$$\dot{V}_{\theta i} \leq -c_i \left\|z_i\right\|^2 \leq -\frac{c_i}{\bar{V}_{\theta i}} V_{\theta i}, \tag{13}$$

where $c_i \triangleq \min\left(\lambda_{\min}(k_{xi}), k_{\theta i}\lambda_{\min}\left(\sum_{\xi=1}^{K}\left(Y_i^{\xi}\right)^T Y_i^{\xi}\right)\right)$. The inequalities in (11) and (13) can then be used to conclude that $\left\|\tilde{x}_i(t)\right\|, \left\|\tilde{\theta}_i(t)\right\| \to 0$ exponentially fast. Thus, the drift dynamics $f_i = Y_i\theta_i^*$ are identified exponentially fast.

Note that even with state derivative estimation errors, the parameter estimation error $\tilde{\theta}_i$ can be shown to be uniformly ultimately bounded (UUB), where the magnitude of the ultimate bound depends on the derivative estimation error [21].

IV. Value Function Approximation

For the approximate value of $\frac{\partial V_i^*}{\partial t} + \mathcal{H}_i\left(\mathcal{E}, \mathcal{X}, \mathcal{U}, V_i^*\right)$ to be evaluated for monitoring purposes, the unknown optimal value function V_i^* needs to be approximated for each agent i. Because the coupled HJB equations are typically infeasible to solve analytically, this section provides an approach to approximate V_i^* using neural networks (NN).

A. Neural network representation

Assuming the networked agents' states remain bounded, the universal function approximation property of NNs can be used with w_i neurons to equivalently represent V_i^* as

$$V_i^*\left(e_i, t'\right) = W_i^T \sigma_i\left(e_i, t'\right) + \epsilon_i\left(e_i, t'\right), \tag{14}$$

where $t' \triangleq \frac{t}{t_f}$ is the normalized time, $W_i \in \mathbb{R}^{w_i}$ is an unknown ideal NN weight vector bounded above by a known constant $\bar{W}_i \in \mathbb{R}_{>0}$ as $\|W_i\| \leq \bar{W}_i$, $\sigma_i : S \times [0,1] \to \mathbb{R}^{w_i}$ is a selected nonlinear, bounded, continuously differentiable activation function with the property $\sigma\left(\underline{0}, 0\right) = \underline{0}$, where $\underline{0}$ is the zero vector of appropriate dimension, and $\epsilon_i : S \times [0,1] \to \mathbb{R}$ is the unknown function reconstruction error. In accordance with the universal function approximation property, the reconstruction error ϵ_i satisfies the properties $\sup_{\varrho \in S, \varphi \in [0,1]} \left|\epsilon_i\left(\varrho, \varphi\right)\right| \leq \bar{\epsilon}_i$, $\sup_{\varrho \in S, \varphi \in [0,1]} \frac{\partial \left|\epsilon_i(\varrho,\varphi)\right|}{\partial \varrho} \leq \bar{\epsilon}_{ei}$ and $\sup_{\varrho \in S, \varphi \in [0,1]} \frac{\partial \left|\epsilon_i(\varrho,\varphi)\right|}{\partial \varphi} \leq \bar{\epsilon}_{ti}$, where $\bar{\epsilon}_i$, $\bar{\epsilon}_{ei}$, $\bar{\epsilon}_{ti} \in \mathbb{R}_{>0}$ are constant upper bounds.

Note that only the state of agent i, states of neighbors of agent i ($j \in \bar{\mathcal{N}}_i$), and time are used as arguments in the NN representation of V_i^* in (14), instead of the states of all agents in the network. This is justified by treating the error states of other agents simply as functions of time, the effect of which is accommodated by including time in the basis function σ_i and function reconstruction error ϵ_i. Inclusion of time in the basis function is feasible due to the finite horizon of the optimization problem in (2). Using state information from additional agents (e.g. two-hop communication) in the network may increase the practical fidelity of function reconstruction and may be done in an approach similar to that developed in this paper.

Using this NN representation, V_i^* is approximated for use in computing the Hamiltonian as

$$\hat{V}_i \triangleq \hat{W}_{ci}^T \sigma_i\left(e_i, t'\right),$$

where $\hat{W}_{ci}$ is an estimate of the ideal NN weight vector W_i.

B. Neural network estimate update policies

To facilitate the development of a feedback-based update policy to drive $\hat{W}_{ci}$ towards W_i, the so-called Bellman error for agent i is defined as

$$\delta_i \triangleq \hat{W}_{ci}^T \sigma_{ti} + \hat{\mathcal{H}}_i\left(E_i, X_i, \hat{W}_{ci}, \hat{\omega}_{ai}, t'\right) - \left(\frac{\partial V_i^*}{\partial t} + \mathcal{H}_i\left(\mathcal{E}, \mathcal{X}, \mathcal{U}^*, V_i^*, t\right)\right), \tag{15}$$

where $\sigma_{ti} \triangleq \frac{\partial \sigma_i}{\partial t}$, $E_i \triangleq \left\{e_j \mid j \in \{i\} \cup \bar{\mathcal{N}}_i\right\}$ is the set of error states of agent i and the neighbors of agent i, $X_i \triangleq \left\{x_j \mid j \in \{i\} \cup \bar{\mathcal{N}}_i\right\}$ is the set of states of agent i and the neighbors of agent i, $\hat{\omega}_{ai} \triangleq \left\{\hat{W}_{ai} \mid j \in \{i\} \cup \bar{\mathcal{N}}_i\right\}$ is the set of actor weights of agent i and the neighbors of agent i, and $\hat{\mathcal{H}}_i$ is the approximate Hamiltonian defined as

$$\hat{\mathcal{H}}_i\left(E_i, X_i, \hat{W}_{ci}, \hat{\omega}_{ai}, t'\right) \triangleq Q_i\left(e_i\right) + \hat{u}_i^T R_i \hat{u}_i + \hat{W}_{ci}^T \sigma_{ei} \times \left(\sum_{j \in \mathcal{N}_i} a_{ij}\left(\hat{f}_i\left(x_i\right) + g_i\hat{u}_i - \hat{f}_j\left(x_j\right) - g_j\hat{u}_j\right) + a_{i0}\left(\hat{f}_i\left(x_i\right) + g_i\hat{u}_i - \dot{x}_0\right)\right), \tag{16}$$

where $\sigma_{ei} \triangleq \frac{\partial \sigma_i}{\partial e_i}$, $\hat{u}_i \triangleq -\frac{1}{2}\left(\sum_{j \in \bar{\mathcal{N}}_i} a_{ij}\right) R_i^{-1} g_i^T \sigma_{ei}^T \hat{W}_{ai}$ is the approximated optimal control for agent i, and $\hat{W}_{ai}$ is another estimate of the ideal NN weight vector W_i. Noting that the expression in (16) is measurable (assuming that the leader state derivative is available to those communicating with the leader), the BE in (15) may be put into measurable form, after recalling that $\frac{\partial V_i^*}{\partial t} + \mathcal{H}_i\left(\mathcal{E}, \mathcal{X}, \mathcal{U}^*, V_i^*\right) \equiv 0$, as

$$\delta_i \triangleq \hat{W}_{ci}^T \sigma_{ti} + \hat{\mathcal{H}}_i\left(E_i, X_i, \hat{W}_{ci}, \hat{\omega}_{ai}, t'\right), \tag{17}$$

which is the feedback to be used to train the NN estimate $\hat{W}_{ci}$. The use of the two NN estimates $\hat{W}_{ci}$ and $\hat{W}_{ai}$ allows for least-squares based adaptation for the feedback in (17), since only the use of $\hat{W}_{ci}$ would result in nonlinearity of $\hat{W}_{ci}$ in (17).

The difficulty in making a non-interfering monitoring scheme with ADP lies in obtaining sufficient data richness for learning. Contrary to typical ADP-based control methods, the developed data-driven adaptive learning policy uses concepts from concurrent learning (cf. [21], [22]) to provide data richness. Let $s_i \triangleq \left\{\varrho_l \in S \mid l = 1, \ldots, \left(\left|\bar{\mathcal{N}}_i\right| + 1\right)\right\} \cup$

$\{\varphi \mid \varphi \in [0, t_f]\}$ be a preselected sample point in the state space of agent i and its neighbors and also time. Additionally, let $S_i \triangleq \{s_i^{c_l} \mid c_l = 1, \ldots, L\}$ be a collection of these unique sample points. The BE will be evaluated over the set S_i in the NN update policies in effort to guarantee data richness. As opposed to the common practice of injecting an exciting signal into a system's control input to provide sufficient data richness for adaptive learning, this strategy evaluates the BE at preselected points in the state space and time to mimic exploration of the state space. The following assumption specifies a sufficient condition on the set S_i for convergence of the subsequently defined update policies.

Assumption 3. For each agent $i \in \mathcal{V}$, the set of sample points S_i satisfies

$$\mu_i \triangleq \frac{1}{L} \inf_{t\in[0,t_f]} \lambda_{\min} \left(\sum_{c_l=1}^{L} \frac{\chi_i^{c_l} \left(\chi_i^{c_l}\right)^T}{\gamma_i^{c_l}} \right) > 0, \tag{18}$$

where $(\cdot)^{c_l}$ denotes evaluation at the c_l^{th} sample point for the indicated agent, $\chi_i \triangleq \sigma_{ti} + \sigma_{ei}\Big(\sum_{j\in\mathcal{N}_i} a_{ij}\left(\hat{f}_i\left(x_i\right) + g_i\hat{u}_i - \hat{f}_j\left(x_j\right) - g_j\hat{u}_j\right) + a_{i0}\left(\hat{f}_i\left(x_i\right) + g_i\hat{u}_i - \dot{x}_0\right)\Big)$ is a regressor vector in the developed NN update law, $\gamma_i \triangleq 1 + \lambda_i \left(\chi_i\right)^T \Gamma_i \chi_i$ provides normalization to the developed NN update law, $\lambda_i \in \mathbb{R}_{>0}$ is a constant normalization gain, and $\Gamma_i \in \mathbb{R}^{w_i \times w_i}$ is a subsequently defined least-squares positive definite gain matrix.

Note that, when performing evaluation of expressions at the preselected concurrent learning sample points, the current values of parameter estimates are used since they are approximating constant values. In general, similar to the selection of a dither signal in PE-based approaches (as in [23], [24]), the satisfaction of (18) cannot be guaranteed a priori. However, this strategy benefits from the ability to accommodate this condition by selecting more information than theoretically necessary; i.e., selecting sample points such that $L \gg w_i$. Additionally, satisfaction of the condition in (18) up until the current time can be verified online.

Using the measurable form of the BE in (17) as feedback, a concurrent learning-based least-squares update policy is developed to approximate W_i as [22]

$$\dot{\hat{W}}_{ci} = -\phi_{c1i}\Gamma_i \frac{\chi_i}{\gamma_i}\delta_i - \frac{\phi_{c2i}}{L}\Gamma_i \sum_{c_l=1}^{L} \frac{\chi_i^{c_l}}{\gamma_i^{c_l}}\delta_i^{c_l}, \tag{19}$$

$$\dot{\Gamma}_i = \left(\beta_i\Gamma_i - \phi_{c1i}\Gamma_i \frac{\chi_i\chi_i^T}{\gamma_i^2}\Gamma_i\right)\mathbf{1}_{\{\|\Gamma_i\| \le \bar{\Gamma}_i\}}, \tag{20}$$

where $\phi_{c1i},\ \phi_{c2i} \in \mathbb{R}_{>0}$ are constant adaptation gains, $\beta_i \in \mathbb{R}_{>0}$ is a constant forgetting factor, $\mathbf{1}_{\{\cdot\}}$ denotes the indicator function, $\bar{\Gamma}_i \in \mathbb{R}_{>0}$ is a saturation constant, and $\Gamma_i(0)$ is positive definite, symmetric, and bounded such that $\|\Gamma_i(0)\| \le \bar{\Gamma}_i$. The form of the least-squares gain matrix update law in (20) is constructed such that Γ_i remains positive definite and

$$\underline{\Gamma}_i \le \|\Gamma_i(t)\| \le \bar{\Gamma}_i, \quad \forall t \in \mathbb{R}_{\ge 0}, \tag{21}$$

where $\underline{\Gamma}_i \in \mathbb{R}_{>0}$ is constant [25]. The NN estimate $\hat{W}_{ai}$ is updated towards the estimate $\hat{W}_{ci}$ as

$$\dot{\hat{W}}_{ai} = -\phi_{a1i}\left(\hat{W}_{ai} - \hat{W}_{ci}\right) - \phi_{a2i}\hat{W}_{ai} + \left(\frac{\phi_{c1i}G_{\sigma i}\hat{W}_{ai}\chi_i^T}{4\gamma_i} + \sum_{c_l=1}^{L} \frac{\phi_{c2i}G_{\sigma i}^{c_l}\hat{W}_{ai}\left(\chi_i^{c_l}\right)^T}{4L\gamma_i^{c_l}}\right)\hat{W}_{ci}, \tag{22}$$

where $\phi_{a1i}, \phi_{a2i} \in \mathbb{R}_{>0}$ are constant adaptation gains, $G_{\sigma i} \triangleq \left(\sum_{j\in\mathcal{N}_i} a_{ij}\right)^2 \sigma_{ei}G_i\sigma_{ei}^T \in \mathbb{R}^{w_i}$, and $G_i \triangleq g_i R_i^{-1} g_i^T$.

C. Stability analysis

Assumption 4. The derivative of the leader state, $\dot{x}_0$, is continuous and bounded.

Theorem 1. *For every agent $i \in \mathcal{V}$, the identifier in (7) along with the adaptive update laws in (19)-(22) guarantee that the estimation errors $\tilde{W}_{ci} \triangleq W_i - \hat{W}_{ci}$ and $\tilde{W}_{ai} \triangleq W_i - \hat{W}_{ai}$ uniformly converge in a finite time T_W to B_r, a ball of radius r centered at the origin, provided Assumptions (1)-(4) hold, the adaptation gains are selected sufficiently large, and the concurrent learning sample points are selected to produce a sufficiently large μ_i, where T_W and r can be made smaller by selecting more concurrent learning sample points to increase the value of μ_i and increasing the gains k_{xi}, $k_{\theta i}$, ϕ_{a1i}, ϕ_{a2i} and ϕ_{c2i}.*

Proof: (Sketch) Consider the Lyapunov function

$$V_L \triangleq \sum_{i\in\mathcal{V}} \left(\frac{1}{2}\tilde{W}_{ci}^T\Gamma_i^{-1}\tilde{W}_{ci} + \frac{1}{2}\tilde{W}_{ai}^T\tilde{W}_{ai} + V_{\theta i}\left(z_i\right)\right). \tag{23}$$

An upper bound of $\dot{V}_L$ along the trajectories of (6), (9), (19) and (22) can be obtained after expressing δ_i in terms of estimation errors, using the property $\frac{d}{dt}\left(\Gamma_i^{-1}\right) = -\Gamma_i^{-1}\dot{\Gamma}_i\Gamma_i^{-1}$, using the inequality $\overline{\left\|\frac{\chi_i}{\gamma_i}\right\|} \le \frac{1}{2\sqrt{\lambda_i\underline{\Gamma}_i}}$, applying the Cauchy-Schwarz and triangle inequalities, and performing nonlinear damping, such that $\dot{V}_L$ is upper-bounded by an expression that is negative definite in terms of the state of V_L plus a constant upper-bounding term. Theorem 4.18 in [26] can then be used to show that the estimation errors are UUB. ■

V. Monitoring Protocol

With the estimation of the unknown NN weight W_i by $\hat{W}_{ci}$ from the previous section, the performance of an agent in satisfying the HJB optimality constraint can be monitored through use of $\hat{V}_i = \hat{W}_{ci}^T\sigma_i$, the approximation of V_i^*. From (4), $\frac{\partial V_i^*}{\partial t} + \mathcal{H}_i\left(\mathcal{E}, \mathcal{X}, \mathcal{U}^*, V_i^*, t\right) \equiv 0$ (where $\mathcal{U}^*$ denotes the optimal control efforts). Let $M_i \in \mathbb{R}$ denote the signal to be monitored by agent $i \in \mathcal{V}$, defined as

$$M_i\left(e_i, X_i, \hat{W}_{ci}, U_i, t'\right) = \Bigg|\hat{W}_{ci}^T\sigma_{ti} + Q_i\left(e_i\right) + u_i^T R_i u_i + \hat{W}_{ci}^T\sigma_{ei}\Bigg(\sum_{j\in\mathcal{N}_i} a_{ij}\left(\hat{f}_i\left(x_i\right) + g_i u_i - \hat{f}_j\left(x_j\right) - g_j u_j\right)$$

$$+ a_{i0}\left(\hat{f}_i\left(x_i\right) + g_i u_i - \dot{x}_0\right)\Bigg)\Bigg|,$$

which differs from (17) in that the measured values of control efforts are being used, where $U_i \triangleq \{u_j \mid j \in \{i\} \cup \mathcal{N}_i\}$. Because the identification of f_i is performed exponentially fast and the uniform convergence of $\tilde{W}_{ci}$ to a ball around the origin occurs in the finite learning time[1] T_W, the monitored signal M_i satisfies the relationship $\left|\frac{\partial V_i^*}{\partial t} + \mathcal{H}_i\left(\mathcal{E}, \mathcal{X}, \mathcal{U}^*, V_i^*, t\right) - M_i\left(e_i, X_i, \hat{W}_{ci}, U_i^*, t'\right)\right| < \varsigma \ \forall t \geq T_W$, where $\varsigma \in \mathbb{R}_{>0}$ is some bounded constant that can be made smaller by selecting larger gains and more concurrent learning sample points. In other words, when using the NN approximation of V_i^* and appropriate gains and sample points, $M_i\left(e_i, X_i, \hat{W}_{ci}, U_i^*, t'\right) \approx 0$, where $U_i^* \triangleq \left\{u_j^* \mid j \in \{i\} \cup \mathcal{N}_i\right\}$. In this manner, due to continuity of the Hamiltonian, observing large values for $M_i\left(e_i, X_i, \hat{W}_{ci}, U_i, t'\right)$ indicates significant deviation away from optimal operating conditions. Let $\bar{M} \in \mathbb{R}_{>0}$ be a constant threshold[2] used for the monitoring process which satisfies $\bar{M} > \varsigma$, where the value for $\bar{M}$ can be increased if a greater tolerance for sub-optimal performance is acceptable. The monitoring protocol, which is separated into a learning phase and a monitoring phase, can be summarized as follows.

Algorithm 1 Monitoring Protocol

Learning phase:
For each agent $i \in \mathcal{V}$, use the update policies in (5), (7), (19), (20), (22) to train $\hat{\theta}_i$ and $\hat{W}_{ci}$, the estimates for θ_i^* and W_i.
Monitoring phase:
At $t = T_W$, terminate updates to $\hat{\theta}_i$ and $\hat{W}_{ci}$.
For each agent $i \in \mathcal{V}$, monitor the value of $M_i\left(e_i, X_i, \hat{W}_{ci}, U_i, t'\right)$ using $\hat{\theta}_i$, $\hat{W}_{ci}$, neighbor communication $\left\{x_j, u_j, \hat{f}_j, g_j\right\}$ and $\dot{x}_0$ if $(i, 0) \in \bar{E}$.
If $M_i > \bar{M}$, then undesirable performance has been observed.

VI. Conclusion

An online decentralized method for monitoring a network of uncertain nonlinear systems in a leader-follower network is presented. The monitoring protocol uses ADP to approximately learn each agent's value function, which is based on the respective performance metric associated with its desired control authority. Concurrent learning is incorporated into the ADP learning policy to guarantee function approximation without the need for injection of a dither signal in the networked dynamical systems, making this a non-intrusive monitoring scheme. The approximated value function is then used to check for satisfaction of the HJB optimality condition so that sub-optimal operating conditions can be recognized.

[1] In practice, determining the precise value for T_W may be difficult; however, T_W may be approximated by observing apparent convergence of the NN weights $\hat{W}_{ci}$.

[2] In practice, similar to other monitoring approaches, trial and error may be used to find an appropriate value for $\bar{M}$.

References

[1] E. Bumiller. (2013, January) Pentagon expanding cybersecurity force to protect networks against attacks. The New York Times.

[2] J. Slay and M. Miller, "Lessons learned from the maroochy water breach," *Crit. Infrastruct. Prot.*, vol. 253, pp. 73–82, 2007.

[3] J. P. Conti, "The day the samba stopped," *Eng. & Technol.*, vol. 5, no. 4, pp. 46–47, 2010.

[4] S. Sundaram, S. Revzen, and G. Pappas, "A control-theoretic approach to disseminating values and overcoming malicious links in wireless networks," *Automatica*, vol. 48, no. 11, pp. 2894–2901, 2012.

[5] W. Abbas and M. Egerstedt, "Securing multiagent systems against a sequence of intruder attacks," in *Proc. Am. Control Conf.*, 2012, pp. 4161–4166.

[6] H. Fawzi, P. Tabuada, and S. Diggavi, "Security for control systems under sensor and actuator attacks," in *Proc. IEEE Conf. Decis. Control*, 2012, pp. 3412–3417.

[7] D. Jung and R. R. Selmic, "Power leader fault detection in nonlinear leader-follower networks," in *Proc. IEEE Conf. Decis. Control*, 2008, pp. 404–409.

[8] X. Zhang, "Decentralized fault detection for a class of large-scale nonlinear uncertain systems," in *Proc. Am. Control Conf.*, 2010, pp. 5650–5655.

[9] Z. Pang and G. Liu, "Design and implementation of secure networked predictive control systems under deception attacks," *IEEE Trans. Control Syst. Tech.*, vol. 20, no. 5, pp. 1334–1342, 2012.

[10] X. Li and K. Zhou, "A time domain approach to robust fault detection of linear time-varying systems," *Automatica*, vol. 45, no. 1, pp. 94–102, 2009.

[11] K. Potula, R. R. Selmic, and M. M. Polycarpou, "Dynamic leader-followers network model of human emotions and their fault detection," in *Proc. IEEE Conf. Decis. Control*, 2010, pp. 744–749.

[12] H. Ferdowsi, D. L. Raja, and S. Jagannathan, "A decentralized fault prognosis scheme for nonlinear interconnected discrete-time systems," in *Proc. Am. Control Conf.*, 2012, pp. 5900–5905.

[13] X. Luo, M. Dong, and Y. Huang, "On distributed fault-tolerant detection in wireless sensor networks," *IEEE Trans. Comput.*, vol. 55, no. 1, pp. 58–70, 2006.

[14] J. Fernández-Bes and J. Cid-Sueiro, "Decentralized detection with energy-aware greedy selective sensors," in *Int. Workshop Cogn. Incr. Process.*, 2012, pp. 1–6.

[15] M. Krstic and Z.-H. Li, "Inverse optimal design of input-to-state stabilizing nonlinear controllers," *IEEE Trans. Autom. Control*, vol. 43, no. 3, pp. 336–350, March 1998.

[16] K. Mombaur, A. Truong, and J.-P. Laumond, "From human to humanoid locomotion - an inverse optimal control approach," *Autonomous Robots*, vol. 28, no. 3, pp. 369–383, 2010.

[17] N. D. Ratliff, J. A. Bagnell, and M. A. Zinkevich, "Maximum margin planning," in *Proc. Int. Conf. Mach. Learn.*, 2006.

[18] R. Beard, G. Saridis, and J. Wen, "Galerkin approximations of the generalized Hamilton-Jacobi-Bellman equation," *Automatica*, vol. 33, pp. 2159–2178, 1997.

[19] D. Kirk, *Optimal Control Theory: An Introduction*. Dover, 2004.

[20] A. Clark, Q. Zhu, R. Poovendran, and T. Başar, "An impact-aware defense against stuxnet," in *Proc. Am. Control Conf.*, 2013, pp. 4146–4153.

[21] G. Chowdhary, T. Yucelen, M. Mühlegg, and E. N. Johnson, "Concurrent learning adaptive control of linear systems with exponentially convergent bounds," *Int. J. Adapt. Control Signal Process.*, vol. 27, no. 4, pp. 280–301, 2013.

[22] R. Kamalapurkar, P. Walters, and W. E. Dixon, "Concurrent learning-based approximate optimal regulation," in *Proc. IEEE Conf. Decis. Control*, Florence, IT, Dec. 2013, pp. 6256–6261.

[23] S. Bhasin, R. Kamalapurkar, M. Johnson, K. Vamvoudakis, F. L. Lewis, and W. Dixon, "A novel actor-critic-identifier architecture for approximate optimal control of uncertain nonlinear systems," *Automatica*, vol. 49, no. 1, pp. 89–92, 2013.

[24] K. Vamvoudakis and F. Lewis, "Online synchronous policy iteration method for optimal control," in *Recent Advances in Intelligent Control Systems*, W. Yu, Ed. Springer, 2009, pp. 357–374.

[25] P. Ioannou and J. Sun, *Robust Adaptive Control*. Prentice Hall, 1996.

[26] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2002.