Single-Agent Indirect Herding of Multiple Targets With Uncertain Dynamics

Ryan A. Licitra, Zachary I. Bell, and Warren E. Dixon, *Fellow, IEEE*

***Abstract*—In this paper, an indirect herding problem is considered for a single herder that is outnumbered by multiple target agents. Indirect herding is a problem that involves a network of one or more controllable herding agents and indirectly controlled target agents (i.e., the target agents can only be controlled through the herder by exploiting herder–target interactions), where the goal is to achieve a network-wide objective. This paper investigates the unique problem where a single herder is required to regulate all the target agents to some desired formation. The problem is further complicated by the fact that the target agents have uncertain nonlinear dynamics, including uncertainties in the herder–target interaction. Neural network function approximation methods are used along with switched systems methods to ensure uniformly ultimately bounded convergence of the agent trajectories provided the developed sufficient dwell-time conditions are satisfied. Simulation and experimental results that involve one herding agent and multiple target agents demonstrate the validity of the designed controller and function approximation scheme for multiple herding objectives.**

***Index Terms*—Adaptive control, Lyapunov methods, multiagent systems, neural networks, nonlinear control systems.**

I. Introduction

VARIOUS applications involve networks of cooperative agents where each agent is directly controlled to achieve a common objective (e.g., consensus, formation control). In leader–follower network problems, leader agents perform actions to direct the network to achieve an objective (which in some cases only the leaders know), whereas followers use centralized or decentralized controllers based on feedback from other agents in the network. Herding and flocking typically refer to network control problems where the leader (or leaders) influences the followers to accomplish an objective (position consensus and velocity vector alignment, respectively) that the leader(s) may not share. Such problems can be considered as *direct* herding problems, where the follower agents are directly controlled. *Indirect* herding problems involve targets that can only be controlled through indirect interaction with a herding agent. The challenges associated with *direct* herding differ greatly from those associated with *indirect* herding problems, and therefore, the remainder of this paper focuses specifically on the *indirect* herding problem; see results such as in [1]–[8] for further details focused on the *direct* herding problem. The distinction between *direct* and *indirect* herding problems, first introduced in [9], is based solely on whether the dynamics of agents being herded in a given problem have an explicit control input (*direct*) or can only be influenced by other agents (*indirect*).

Indirect herding can include noncooperative agents, similar to pursuer–evader problems (cf., [10]–[16]). Several results have been published that focus on a herding version of the pursuer–evader problem (cf., [17]–[21]), where pursuit–evasion problems typically terminate upon capture, and herding pursuer–evader problems involve both capture and regulation to a desired location.

Motivated by the foundational work based on observations from nature and experimental development in [22]–[26], various *indirect* herding approaches have been investigated. In [17], the *indirect* herding pursuer–evader problem is developed as an online optimal control problem for a known dynamic system consisting of a single pursuer and single evader. Results, such as in [18]–[20], solve the same problem with additional pursuers from a game theoretic perspective using offline numerical solutions. In [27], the position of multiple herders equidistant from the target are used to define a forcing function to guide the target along a desired trajectory. A similar arc-based approach is used in [28] to regulate multiple targets to a desired goal, treating the entire herd as a single unicycle centered at the mean location of the herd.

In this paper, an approach to solving the *indirect* herding problem is developed for a single herder and multiple target agents. Since the herder is outnumbered by the targets, a strategy is required to switch between different targets, unlike results such as in [17]–[19], [20], [27], and [28]. Switched systems analysis methods are used to develop sufficient dwell-time conditions that inform the herder of the minimum time that it must interact with an agent before it can switch to another target. Our previous work in [29] solves a similar outnumbered herding

Manuscript received January 2, 2019; accepted March 15, 2019. Date of publication May 30, 2019; date of current version August 1, 2019. This work was supported in part by a Task Order contract with the Air Force Research Laboratory, Munitions Directorate, at Eglin AFB, Office of Naval Research, under Grant N00014-13-1-0151, in part by the Air Force Office of Scientific Research (AFOSR) under Grant FA9550-18-1-0109 and Grant FA9550-19-1-0169, and in part by the NEEC under Grant N00174-18-1-0003. This paper was recommended for publication by Associate Editor M. Schwager and Editor P. Robuffo Giordano upon evaluation of the reviewers' comments. *(Corresponding author: Ryan A. Licitra.)*

The authors are with the Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL 32611-6250 USA (e-mail: rlicitra@ufl.edu; bellz121@ufl.edu; wdixon@ufl.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors. The material consists of a video, viewable with Windows Media Player (version 12.0.16299.248 or newer) or VLC Media Player (version 2.2.8 or newer), demonstrating a typical run of Experiments 1 and 2 from the attached manuscript. The size of the video is 36.3 MB. Contact rlicitra@ufl.edu for further questions about this work.

Digital Object Identifier 10.1109/TRO.2019.2911799

problem from a robust control perspective using a sliding-mode controller (SMC) to compensate for uncertainties in the model and serves as a stepping stone result for this paper. More recently, our work in [9] relaxes the need for the high-frequency SMC by exploiting an adaptive control approach, which takes advantage of the linear-in-the-parameters assumption about the dynamic model, using an integral concurrent learning (ICL) [30], [31] scheme to improve parameter estimation and facilitate the switched systems analysis. In both [9] and [29], the herder–target interaction was assumed to have a specific potential function structure. In this paper, a more general model is considered that only assumes the target has some repulsion from the herder as a function of the distance between the herder and the target. A neural network (NN) is used to approximate the unknown herder–target interaction, where ICL is incorporated within the NN update law to yield a quantifiable transient performance that is essential to develop dwell-time conditions for switching. The ICL scheme requires only a finite excitation (FE) condition for estimating the unknown dynamics of the herder and targets, which can be verified online (as opposed to the persistence of excitation (PE) condition often used in the adaptive control literature, which is difficult to satisfy and cannot be verified in general for nonlinear systems).

Since the target agent dynamics are not directly controllable, the interaction between the targets and herder (i.e., the unknown function of the distance between the target and the herder) is exploited to exert influence on the targets. Specifically, integrator backstepping [32, Ch. 2] is used to inject a virtual controller in the target dynamics. A Lyapunov-based switched systems analysis is used to guarantee that all targets are regulated to a neighborhood of the goal location, and once the FE condition is satisfied, the targets are proven to converge to an even smaller neighborhood.

In addition to the stability analysis, the performance of the developed controller is demonstrated through experimental results. The experiments use a quadcopter as an example herding agent and six paper targets of different heights and weights are used as target agents. The paper objects are used to highlight the fact that the targets can only be indirectly controlled through uncertain aerodynamic forces exerted on the paper objects by the quadcopter. In one experiment, the quadcopter herds all the paper objects to a desired common region, and in another experiment, the quadcopter sorts the targets into different regions based on the color of the targets. For each experiment, a specific strategy is invoked to determine the order of targets that are regulated; however, the mathematical development is agnostic to the specific strategy, as long as it satisfies the developed dwell-time conditions.

This paper is organized as follows. Section II describes the agents' dynamic models and introduces the problem, whereas Sections III and IV contain the control objective and control development, respectively. In the latter, both the control design and the function approximation scheme are provided. Section V includes the multistep stability analysis, including the switched systems analysis that makes the *indirect* herding problem possible when there are fewer herders than targets. Simulation and experimental results are provided in Section VI, with thorough discussions in Section VII, and finally Section VIII concludes this paper.

II. Problem Formulation

Consider a network consisting of one herding agent, denoted by $y \in \mathbb{R}^n$ (where n is the dimensionality of the system), and $n_t \in \mathbb{Z}_{>0}$ target agents. Let $\mathcal{T} \triangleq \{1, 2, ..., n_t\}$ be the set of target agents. The herding agent is tasked with regulating the targets, each of whose position is denoted by $x_i \in \mathbb{R}^n$, to their respective constant goal locations, $x_{i,g} \in \mathbb{R}^n \quad \forall\, i \in \mathcal{T}$. The target, goal, and herder states are assumed to be available to the herder for feedback control at all times. In contrast to traditional leader–follower network problems where each agent's interaction is controlled to accomplish a common goal, in the *indirect* herding problem, only the herder's dynamics are directly controllable, whereas the target states are only influenced through interaction with the herder. The herder dynamics are given by

$$\dot{y} = h\left(x, y\right) + u_y \tag{1}$$

where $x = \left[\, x_1^T \;\; x_2^T \;\; ... \;\; x_{n_t}^T \,\right]^T$ is a stacked vector of the target states, $h : \mathbb{R}^n \times \mathbb{R}^{nn_t} \to \mathbb{R}^n$ is an unknown locally Lipschitz function that represents the potentially nonlinear herder dynamics, and $u_y \in \mathbb{R}^n$ is the herder controller. The target agents are influenced through an unknown repulsion that is a function of the distance between the target and the herder. Specifically, the ith target's potentially nonlinear motion model is given by

$$\dot{x}_i = \alpha_i \left(\|x_i - y\|\right)\left(x_i - y\right) + f_i\left(x, y\right) \tag{2}$$

where $\alpha_i : \mathbb{R} \to \mathbb{R}$ is an unknown locally Lipschitz function that is bounded by

$$\underline{\alpha}_i \leq \alpha_i \left(\|x_i - y\|\right) \leq \bar{\alpha}_i$$

where $\underline{\alpha}_i, \bar{\alpha}_i,\ i \in \mathcal{T}$ are known[1] positive constants, and $f_i : \mathbb{R}^n \to \mathbb{R}^n$ is an unknown bounded,[2] locally Lipschitz function that defines any additional ith target dynamics. For example, these additional dynamics could dictate that the target group around other targets or flee its goal location. The first term in (2) models the repulsion interaction between the herder and the ith target.

Since a single herder is tasked with controlling multiple targets, a switching strategy is employed to ensure that the herding task is accomplished (i.e., all targets are regulated to their goal location). The herding agent switches between chasing each target toward its goal location, one at a time, according to a switching strategy that must satisfy subsequently developed sufficient dwell-time conditions. At any given time,[3] a target will either be operating in a *chased* or *unchased* mode. Let $t_{i,k}^c \in \mathbb{R}$ and $t_{i,k}^u \in \mathbb{R}$ denote the kth instance when the ith target is switched to the *chased* or *unchased* mode, respectively, where $k \in \mathbb{N}$.

[1] A reasonable understanding of the bounds on the unknown dynamics is required to check the subsequently developed sufficient conditions.

[2] The target drift dynamics f_i must be upper bounded either by a constant or by a function of the target states.

[3] Exactly one target will be in the *chased* mode, while all others operate in the *unchased* mode, at any given time.

The contiguous dwell time in the kth activation of the ith target operating in the *chased* or *unchased*mode is denoted by $\Delta t_{i,k}^c \in \mathbb{R}$ and $\Delta t_{i,k}^u \in \mathbb{R}$, and defined as $\Delta t_{i,k}^c \triangleq t_{i,k}^u - t_{i,k}^c$ and $\Delta t_{i,k}^u \triangleq t_{i,k+1}^c - t_{i,k}^u$, respectively. The total amount of time for which each of these modes is active between switching instances a and b is denoted by $T_i^c(a,b) \triangleq \sum_{l=a}^{b} \Delta t_{i,l}^c$ and $T_i^u(a,b) \triangleq \sum_{l=a}^{b} \Delta t_{i,l}^u$, respectively. To facilitate the subsequent analysis in Section V, the target currently operating in the *chased* mode is denoted by adding the superscript c (i.e., $x_i(t) = x_i^c(t) \quad \forall t \in [t_{i,k}^c, t_{i,k}^u) \quad \forall k \in \mathbb{N}$).

III. Control Objective

The objective is to control the herder such that it regulates all target agents to their goal locations, despite the uncertainties in the system and the target's uncertain noncooperative behavior. To quantify the herding objective, the target regulation error, denoted by $\bar{x}_i \in \mathbb{R}^n$, is defined as

$$\bar{x}_i(t) \triangleq x_i(t) - x_{i,g}. \tag{3}$$

Since both (1) and (2) contain unknown dynamics, function approximation methods are employed. The ith target dynamics in (2) and the unknown dynamics in (1) can each be approximated by an NN [33] as[4]

$$\alpha_i(x_i - y) + f_i = W_i^T \sigma_i(\zeta) + \varepsilon_i(\zeta) \tag{4}$$

and[5]

$$-h = W_y^T \sigma_y(\zeta) + \varepsilon_y(\zeta) \tag{5}$$

respectively, where $\zeta \in \mathbb{R}^{n(n_t+1)}$ is defined as $\zeta \triangleq \begin{bmatrix} x^T & y^T \end{bmatrix}^T$, and $\sigma_i : \mathbb{R}^{n(n_t+1)} \to \mathbb{R}^{L_1}$ and $\sigma_y : \mathbb{R}^{n(n_t+1)} \to \mathbb{R}^{L_2}$ are known, bounded, locally Lipschitz, vectors of basis functions. Moreover, $W_i \in \mathbb{R}^{L_1 \times n}$ and $W_y \in \mathbb{R}^{L_2 \times n}$ are matrices of the unknown ideal weights, $L_1 \in \mathbb{N}$ is the number of neurons used in the NN in (4), $L_2 \in \mathbb{N}$ is the number of neurons used in the NN in (5), and $\varepsilon_i : \mathbb{R}^{n(n_t+1)} \to \mathbb{R}^n$ and $\varepsilon_y : \mathbb{R}^{n(n_t+1)} \to \mathbb{R}^n$ are the function approximation residual errors.

Remark 1: The function approximation residual errors can be upper bounded by positive constants that can be made arbitrarily small based on the Stone–Weierstrass theorem [34], i.e., $\bar{\varepsilon}_i \triangleq \sup_{\zeta \in \chi, t \in [0,\infty)} \|\varepsilon_i(\zeta(t))\| \quad \forall i \in \mathcal{T}$, and $\bar{\varepsilon}_y \triangleq \sup_{\zeta \in \chi, t \in [0,\infty)} \|\varepsilon_y(\zeta(t))\|$. The Stone–Weierstrass theorem requires that the states remain in a compact set (i.e., $\zeta(t) \in \chi$). The stability proof in Section V shows that if $\zeta(0)$ is bounded, then $\zeta(t) \in \chi$, where χ is a compact simply connected set such that $\chi \subset \mathbb{R}^{n(n_t+1)}$.

Let

$$\tilde{W}_i(t) \triangleq W_i - \hat{W}_i(t)$$

and

$$\tilde{W}_y(t) \triangleq W_y - \hat{W}_y(t)$$

denote the parameter estimation error for the weights associated with the ith target and the herder, respectively, where $\hat{W}_i \in \mathbb{R}^{L_1 \times n}$ is the estimate of the ideal function approximation weights associated with the ith target, and $\hat{W}_y \in \mathbb{R}^{L_2 \times n}$ is the estimate of the ideal function approximation weights associated with the herder.

Based on (4), the dynamics in (2) can be rewritten as

$$\dot{x}_i = W_i^T \sigma_i + \varepsilon_i \tag{6}$$

while (1) can be rewritten using (5) as

$$\dot{y} = -\left(W_y^T \sigma_y + \varepsilon_y\right) + u_y. \tag{7}$$

Given that the target agent dynamics in (2) do not explicitly contain a control input, a backstepping strategy is used to inject the desired herder state as a virtual controller $y_d \in \mathbb{R}^n$ into the dynamics of the target currently operating in the *chased* mode. Therefore, in addition to the target regulation error in (3), another objective is to minimize the backstepping error $e_y \in \mathbb{R}^n$ defined as

$$e_y(t) \triangleq y_d(t) - y(t). \tag{8}$$

Since the desired herder position $y_d(t)$ is defined such that the herding objective is achieved for the currently chased target, the backstepping error (8) is introduced as a regulation objective in the control development in Section IV. With the inclusion of (8) in the control development and then in the stability analysis in Section V, the herder is ensured to be driven to the desired herder position, which in turn ensures that the herding objective is accomplished.

IV. Control Development

The herder is tasked with leveraging its unknown physical influence on each target such that all targets are relocated to their associated goal location. In Section IV-A, a control law is developed for the herder to move such that a single target is regulated to its goal position, made possible by the NN-based function approximation detailed in Section IV-B. This one herder versus one target development is later used in conjunction with a switched systems framework in the stability analysis in Section V to show how the overall herding objective is achieved.

A. Herding Controller

In the following development, the herder switches between chasing each target to achieve the overall herding objective, and thus, the controller always uses the currently *chased* target (only one target at a time can operate as the *chased* target). To this end, the notation x_i^c denotes the target that is currently operating in the *chased* mode (i.e., $x_i(t) = x_i^c(t) \; \forall t \in \left[t_{i,k}^c, t_{i,k}^u\right) \; \forall k \in \mathbb{N}$) and $x_{i,g}^c$ refers to the goal position associated with the same target (and all other terms with superscript c follow suit). To develop the controller, the target dynamics in (2), and the backstepping error in (8) are used to express the time derivative of (3) as

$$\dot{\bar{x}}_i^c = \alpha_i^c\left(x_i^c + e_y - y_d\right) + f_i^c. \tag{9}$$

[4]For notational brevity, functional dependence on states and time will be henceforth suppressed, except for when introducing new terms and where necessary for clarity.

[5]Due to the way that $e_y(t)$ is later defined, the negative sign is included on the left-hand side of (5) to avoid a potentially confusing negative sign in the Lyapunov analysis in Section V.

Based on the desire to regulate the herding error, the herder's desired state is designed as

$$y_d \triangleq K_{1,i}^c \bar{x}_i^c + x_{i,g}^c \tag{10}$$

where[6] $K_{1,i} = k_{1,i} + k_{2,i} + k_{n,i}$ and $k_{1,i}, k_{2,i}, k_{n,i}$ are positive constant control gains $\forall i \in \{1, 2, ..., n_t\}$. Using (10), (9) can be rewritten as

$$\dot{\bar{x}}_i^c = \alpha_i^c \left(1 - K_{1,i}^c\right) \bar{x}_i^c + \alpha_i^c e_y + f_i^c. \tag{11}$$

The backstepping error dynamics can be determined by taking the time derivative of (8), and using (1), (2), (4), and (5) to obtain

$$\dot{e}_y = K_{1,i}^c \left(\left(W_i^c\right)^T \sigma_i^c + \varepsilon_i^c\right) + W_y^T \sigma_y + \varepsilon_y - u_y. \tag{12}$$

Based on (12) and the subsequent stability analysis in Section V, the herder control law is designed as

$$u_y \triangleq K_{2,i}^c e_y + K_{1,i}^c \left(\hat{W}_i^c\right)^T \sigma_i^T + \hat{W}_y^T \sigma_y + k_{s,i}^c \mathrm{sgn}\left(e_y\right) \tag{13}$$

where $\forall i \in \{1, 2, ...n_t\}, K_{2,i} = k_y + k_{3,i}, k_y, k_{3,i}, k_{s,i}$ are positive constant control gains, and $\mathrm{sgn}\left(\cdot\right)$ is the signum function. Using (13), the closed-loop backstepping dynamics in (12) can be rewritten as

$$\begin{aligned}\dot{e}_y = {} & K_{1,i}^c \left(\tilde{W}_i^c\right)^T \sigma_i^c + \tilde{W}_y^T \sigma_y + K_{1,i}^c \varepsilon_i^c \\ & + \varepsilon_y - K_{2,i}^c e_y - k_{s,i}^c \mathrm{sgn}\left(e_y\right).\end{aligned} \tag{14}$$

B. Function Approximation

This section details the function approximation scheme necessary for the implementation of the controller designed in Section IV-A. The following steps are useful to facilitate the development of the FE condition associated with the ICL strategy that is introduced in the following.

Taking the transpose of (7) and integrating yields

$$\begin{aligned}\int_{t-\Delta t}^{t} \dot{y}^T\left(\varsigma\right) d\varsigma = {} & -\int_{t-\Delta t}^{t} \sigma_y^T\left(y\left(\varsigma\right)\right) W_y d\varsigma \\ & - \int_{t-\Delta t}^{t} \varepsilon_y^T\left(y\left(\varsigma\right)\right) d\varsigma \\ & + \int_{t-\Delta t}^{t} u_y^T\left(\varsigma\right) d\varsigma\end{aligned} \tag{15}$$

where $\Delta t \in \mathbb{R}$ is a positive constant denoting the size of the window of integration. Using the fundamental theorem of calculus, (15) can be rewritten as

$$\mathcal{U}_y\left(t\right) - y^T\left(t\right) + y^T\left(t - \Delta t\right) = \mathcal{Y}_y\left(t\right) W_y + \mathcal{E}_y\left(t\right) \tag{16}$$

$\forall t \in [\Delta t, \infty)$, where $\mathcal{Y}_y\left(t\right) \triangleq \int_{t-\Delta t}^{t} \sigma_y^T\left(y\left(\varsigma\right)\right) d\varsigma$, $\mathcal{E}_y\left(t\right) \triangleq \int_{t-\Delta t}^{t} \varepsilon_y^T\left(y\left(\varsigma\right)\right) d\varsigma$, and $\mathcal{U}_y\left(t\right) \triangleq \int_{t-\Delta t}^{t} u_y^T\left(\varsigma\right) d\varsigma$. Similarly, taking the transpose of (6) and integrating yields

$$\begin{aligned}\int_{t-\Delta t}^{t} \dot{x}_i^T\left(\varsigma\right) d\varsigma = {} & \int_{t-\Delta t}^{t} \sigma_i^T\left(\zeta_i\left(\varsigma\right)\right) W_i d\varsigma \\ & + \int_{t-\Delta t}^{t} \varepsilon_i^T\left(\zeta_i\left(\varsigma\right)\right) d\varsigma.\end{aligned} \tag{17}$$

Using the fundamental theorem of calculus, (17) can be rewritten as

$$x_i^T\left(t\right) - x_i^T\left(t - \Delta t\right) = \mathcal{Y}_i\left(t\right) W_i + \mathcal{E}_i\left(t\right) \quad \forall t \in [\Delta t, \infty) \tag{18}$$

where $\forall t \in [\Delta t, \infty)$, $\mathcal{Y}_i\left(t\right) \triangleq \int_{t-\Delta t}^{t} \sigma_i^T\left(\zeta_i\left(\varsigma\right)\right) d\varsigma$, and $\mathcal{E}_i\left(t\right) \triangleq \int_{t-\Delta t}^{t} \varepsilon_i^T\left(\zeta_i\left(\varsigma\right)\right) d\varsigma$.

The parameter estimate for the weights associated with the herder [i.e., $\hat{W}_y$ in (13)] is generated from the ICL-based adaptive update law [30], [31]

$$\begin{aligned}\dot{\hat{W}}_y \triangleq \mathrm{proj}\Bigg\{ & \Gamma_y \sigma_y e_y^T + k_{\mathrm{CL},y} \Gamma_y \\ & \times \sum_{j=1}^{N_y} \mathcal{Y}_{y,j}^T \left(\mathcal{U}_{y,j} - \Delta y_j - \mathcal{Y}_{y,j} \hat{W}_y\right)\Bigg\}\end{aligned} \tag{19}$$

where $\mathrm{proj}\left\{\cdot\right\}$ is a smooth projection operator,[7] $\Gamma_y \in \mathbb{R}^{L_2 \times L_2}$ and $k_{\mathrm{CL},y} \in \mathbb{R}$ are constant, positive definite control gains, $\Delta y_j \triangleq y^T\left(t_j\right) - y^T\left(t_j - \Delta t\right)$, $N_y \in \mathbb{Z}$ is a constant that represents the number of saved data points for the data stack of the herder, and $t_j \in [\Delta t, t]$ represents time points when measurements are available. Likewise, the parameter estimate for the weights associated with the ith target (i.e., $\hat{W}_i$) is generated from the ICL-based adaptive update law

$$\begin{aligned}\dot{\hat{W}}_i \triangleq \mathrm{proj}\Bigg\{ & \Gamma_i K_{1,i} \sigma_i e_y^T + k_{\mathrm{CL},i} \Gamma_i \\ & \times \sum_{j=1}^{N_i} \mathcal{Y}_{i,j}^T \left(\Delta x_{i,j} - \mathcal{Y}_{i,j} \hat{W}_i\right)\Bigg\}\end{aligned} \tag{20}$$

where $\forall i \in \{1, 2, ..., n_t\}$, $\Gamma_i \in \mathbb{R}^{L_1 \times L_1}$, and $k_{\mathrm{CL},i} \in \mathbb{R}$ are constant, positive definite control gains, $\Delta x_{i,j} \triangleq x_i^T\left(t_j\right) - x_i^T\left(t_j - \Delta t\right)$, $N_i \in \mathbb{Z}$ is a constant that represents the number of saved data points for the data stack of the ith target, and $t_j \in [\Delta t, t]$ represents time points when measurements are available.

The idea behind concurrent learning [36] is to utilize recorded input and output data to identify the ideal weights. The data points that are saved are selected to maximize the minimum eigenvalues of $\sum_{j=1}^{N_y} \left(\mathcal{Y}_{y,j}^T \mathcal{Y}_{y,j}\right)$ and $\sum_{j=1}^{N_i} \left(\mathcal{Y}_{i,j}^T \mathcal{Y}_{i,j}\right)$[8]. Using (16) and (18), the adaptive update laws in (19) and (20) can be rewritten in the following equivalent, but nonimplementable,[9]

[6]The gain parameter $K_{1,i}$ is a sum of positive gains for notational convenience in the stability analysis. In practice, this is implemented as a single quantity.

[7]The limits used in the projection algorithm are based on the known bounds on the unknown functions in (1) and (2). See [35, Sec. 4.4] for details of the projection operator.

[8]See [37] for details on methods of selecting data.

[9]The update laws (21) and (22) contain $\tilde{W}_y$ and $\tilde{W}_i$, respectively, which are unknown.

forms, respectively:

$$\dot{\hat{W}}_y = \text{proj}\Bigg\{\Gamma_y \sigma_y e_y^T + k_{\text{CL},y}\Gamma_y \sum_{j=1}^{N_y} \mathcal{Y}_{y,j}^T \mathcal{Y}_{y,j}\tilde{W}_y + k_{\text{CL},y}\Gamma_y \sum_{j=1}^{N_y} \mathcal{Y}_{y,j}^T \mathcal{E}_{y,j}\Bigg\} \tag{21}$$

$$\dot{\hat{W}}_i = \text{proj}\Bigg\{\Gamma_i K_{1,i}\sigma_i e_y^T + k_{\text{CL},i}\Gamma_i \sum_{j=1}^{N_i} \mathcal{Y}_{i,j}^T \mathcal{Y}_{i,j}\tilde{W}_i + k_{\text{CL},i}\Gamma_i \sum_{j=1}^{N_i} \mathcal{Y}_{i,j}^T \mathcal{E}_{i,j}\Bigg\} \tag{22}$$

for all $t > \Delta t$, where $\mathcal{E}_{y,j} \triangleq \mathcal{E}_y(t_j)$ and $\mathcal{E}_{i,j} \triangleq \mathcal{E}_i(t_j)$. Additionally, during periods that the ith target is operating in the *unchased* mode, the gradient term in (20) is turned OFF, i.e., $\forall t \in [t_{i,k}^u, t_{i,k+1}^c) \quad \forall k \in \mathbb{N}$,

$$\dot{\hat{W}}_i \triangleq \text{proj}\left\{k_{\text{CL},i}\Gamma_i \sum_{j=1}^{N_i} \mathcal{Y}_{i,j}^T \left(\Delta x_{i,j} - \mathcal{Y}_{i,j}\hat{W}_i\right)\right\} \tag{23}$$

which can be rewritten as

$$\dot{\hat{W}}_i = \text{proj}\left\{k_{\text{CL},i}\Gamma_i \sum_{j=1}^{N_i} \mathcal{Y}_{i,j}^T \mathcal{Y}_{i,j}\tilde{W}_i + k_{\text{CL},i}\Gamma_i \sum_{j=1}^{N_i} \mathcal{Y}_{i,j}^T \mathcal{E}_{i,j}\right\}. \tag{24}$$

V. Stability Analysis

This stability analysis considers the behavior of the ith target when it is in the *chased* and *unchased* modes, and then a combined switched systems analysis is provided to quantify the behavior of the overall state trajectories. To facilitate the analysis, the time periods before and after the FE condition is satisfied are considered. The following analysis assumes that the following FE[10] conditions are satisfied [30], [31]

$$\exists \underline{\lambda}_y > 0, \tau_y > \Delta t : \forall t \geq \tau_y,\ \lambda_{\min}\left\{\sum_{j=1}^{N_y} \mathcal{Y}_{y,j}^T \mathcal{Y}_{y,j}\right\} \geq \underline{\lambda}_y \tag{25}$$

$$\exists \underline{\lambda}_i > 0, \tau_i > \Delta t : \forall t \geq \tau_i,\ \lambda_{\min}\left\{\sum_{j=1}^{N_i} \mathcal{Y}_{i,j}^T \mathcal{Y}_{i,j}\right\} \geq \underline{\lambda}_i \tag{26}$$

where $\lambda_{\min}\{\cdot\}$ refers to the minimum eigenvalue of $\{\cdot\}$.

The stability analysis is developed as follows. Lemma 1 proves that during periods when a target operates in the *chased* mode, the system states associated with the ith target decay asymptotically prior to sufficient excitation (i.e., $t \in [0, \tau_i)$) and are exponentially stable after sufficient excitation (i.e., $t \in [\tau_i, \infty)$). Lemma 2 shows that the target states remain bounded for all bounded t when the ith target is operating in the *unchased* mode. Considering these results, the overall trajectories are analyzed in Theorems 1 and 2, and ultimate bounds are provided for the system states associated with the ith target during the two time periods, respectively, provided that the sufficient gain and dwell-time conditions are met.

To facilitate the following analysis, let $V_i : \mathbb{R}^{2n+nL_1+nL_2} \to \mathbb{R}$ be a positive definite, continuously differentiable candidate Lyapunov function, defined as

$$V_i(z_i(t)) \triangleq \frac{1}{2}\bar{x}_i^T \bar{x}_i + \frac{1}{2}e_y^T e_y + \frac{1}{2}\text{tr}\left(\tilde{W}_i^T \Gamma_i^{-1}\tilde{W}_i\right) + \frac{1}{2}\text{tr}\left(\tilde{W}_y^T \Gamma_y^{-1}\tilde{W}_y\right) \tag{27}$$

which can be bounded as

$$\beta_1 \|z_i(t)\|^2 \leq V_i(z_i(t)) \leq \beta_2 \|z_i(t)\|^2 \tag{28}$$

where $z_i(t) \in \mathbb{R}^{2n+nL_1+nL_2}$ is the stacked state vector, defined as

$$z_i \triangleq \left[\bar{x}_i^T \ e_y^T \ \text{vec}\left(\tilde{W}_i\right)^T \ \text{vec}\left(\tilde{W}_y\right)^T\right]^T$$

$\text{tr}(\cdot)$ denotes the matrix trace operator, $\text{vec}(\cdot)$ denotes a stack of the columns of $(\cdot)$, and $\beta_1, \beta_2 \in \mathbb{R}$ are known positive bounding constants. Moreover, since the use of the projection algorithm in (19) and (20) ensures that $\tilde{W}_y, \hat{W}_y, \tilde{W}_i, \hat{W}_i \in \mathcal{L}_\infty$, then the Lyapunov function candidate can also be upper bounded as

$$V_i(z_i(t)) \leq \beta_3 \left\|\left[\bar{x}_i^T \ e_y^T\right]\right\|^2 + \beta_4 \tag{29}$$

where $\beta_3, \beta_4 \in \mathbb{R}$ are known positive bounding constants.

A. Target Operating in the Chased Mode

In this section, the ith target is the one currently operating in the *chased* mode; however, for clarity in this analysis, the subscript i will be used in lieu of the subscript c. This notation is used to avoid confusion in the combined analysis in Section V-C, where the state trajectories associated with the ith target are considered over all time, switching between the *chased* and *unchased* modes.

The following Lemma establishes stability of the ith target during periods in which it operates in the *chased* mode.

Lemma 1: The controller given in (10) and (13), and the adaptive update laws in (19) and (20) ensure that all system signals associated with the ith target are bounded under closed-loop operation and that $\forall t \in [t_{i,k}^c, t_{i,k}^u) \quad \forall k \in \mathbb{N}$,

$$\|z_i(t)\|^2 \leq \frac{\beta_2}{\beta_1}\left\|z_i\left(t_{i,k}^c\right)\right\|^2 \text{e}^{-\lambda_{1,i}\left(t - t_{i,k}^c\right)} + \frac{\kappa_{1,i}}{\beta_1} \tag{30}$$

where $\kappa_{1,i} \in \mathbb{R}$ is a subsequently defined known positive constant, provided that the gains are selected according to the

[10]The condition in (26) requires that the system be sufficiently excited, which is a milder (can be satisfied in finite time τ_i) condition than the typical PE condition. For more information about the FE condition and how likely a given system is to satisfy it, see [36], [38], and [39].

sufficient gain conditions

$$k_{2,i} \geq \frac{3\bar{\alpha}_i}{2\underline{\alpha}_i} \tag{31}$$

$$k_{3,i} \geq \frac{\bar{\alpha}_i}{2} \tag{32}$$

$$k_{s,i} \geq c_{NN,i} \tag{33}$$

where $c_{\text{NN},i} \in \mathbb{R}$ is a known positive constant that upper bounds the residual function approximation error. Moreover, provided the inequalities in (25) and (26) are satisfied (i.e., the trajectories are sufficiently exciting), then $\forall t \in \left[t_{i,k}^c, t_{i,k}^u\right) \cap [\tau_i, \infty) \quad \forall k \in \mathbb{N}$

$$\|z_i(t)\|^2 \leq \frac{\beta_2}{\beta_1} \left\|z_i\left(t_{i,k}^c\right)\right\|^2 \mathrm{e}^{-\lambda_{2,i}\left(t-t_{i,k}^c\right)} + \frac{\kappa_{2,i}}{\beta_1} \tag{34}$$

where $\kappa_{2,i} \in \mathbb{R}$ is a subsequently defined known positive constant.

Proof: Using (11), (14), (21), and (22), the time derivative of (27) during $t \in [t_{i,k}^c, t_{i,k}^u) \quad \forall k \in \mathbb{N}$ can be upper bounded as

$$\begin{aligned}\dot{V}_i \leq & -k_{1,i}\underline{\alpha}_i \|\bar{x}_i\|^2 - k_y \|e_y\|^2 - \left(k_{2,i} - \frac{3\bar{\alpha}_i}{2\underline{\alpha}_i}\right)\|\bar{x}_i\|^2 \\ & - \left(k_{3,i} - \frac{\bar{\alpha}_i}{2}\right)\|e_y\|^2 + (c_{\text{NN},i} - k_{s,i})\|e_y\| \\ & + \frac{\bar{f}_i}{4k_{n,i}\underline{\alpha}_i} + c_{\text{CL},i}\end{aligned} \tag{35}$$

where $c_{\text{CL},i} \in \mathbb{R}$ is a known positive constant. Note that in (35), the terms containing the NN weight estimation errors are both upper bounded by zero since they are only negative semidefinite during the learning phase (i.e., before enough data have been collected). Provided the sufficient gain conditions in (31)–(33) are satisfied, (35) can be upper bounded as

$$\dot{V}_i(z_i(t)) \leq -k_{1,i}\underline{\alpha}_i \|\bar{x}_i\|^2 - k_y \|e_y\|^2 + C_{1,i} \tag{36}$$

where $C_{1,i} \triangleq \frac{\bar{f}_i}{4k_{n,i}\underline{\alpha}_i} + c_{\text{CL},i}$. Using (29), (36) can be upper bounded as

$$\dot{V}_i(z_i(t)) \leq -\lambda_{1,i} V_i(z_i(t)) + \lambda_{1,i}\beta_4 + C_{1,i} \tag{37}$$

where $\lambda_{1,i} \triangleq \frac{1}{\beta_3}\min\{k_{1,i}\underline{\alpha}_i, k_y\}$. Applying the comparison lemma [40, Lemma 3.4] to (37) yields

$$V_i(z_i(t)) \leq V_i\left(z_i\left(t_{i,k}^c\right)\right)\mathrm{e}^{-\lambda_{1,i}\left(t-t_{i,k}^c\right)} + \kappa_{1,i} \tag{38}$$

$\forall t \in [t_{i,k}^c, t_{i,k}^u) \quad \forall k \in \mathbb{N}$, where $\kappa_{1,i} \triangleq (\beta_4 + \frac{C_{1,i}}{\lambda_{1,i}})$. Using (28), (30) can be obtained.

Once sufficient data have been collected (i.e., $t \in [t_{i,k}^c, t_{i,k}^u) \cap [\tau_y, \infty) \cap [\tau_i, \infty)$) and provided that the sufficient gain conditions (31)–(33) are satisfied, the time derivative of (27) can be upper bounded as

$$\begin{aligned}\dot{V}_i \leq & -k_{1,i}\underline{\alpha}_i \|\bar{x}_i\|^2 - k_y \|e_y\|^2 \\ & - k_{\text{CL},y}\underline{\lambda}_y \left\|\tilde{W}_y\right\|^2 - k_{\text{CL},i}\underline{\lambda}_i \left\|\tilde{W}_i\right\|^2 + C_{1,i}.\end{aligned}$$

Furthermore, (28) can be used to show that

$$\dot{V}_i(z_i(t)) \leq -\lambda_{2,i} V_i(z_i(t)) + C_{1,i} \tag{39}$$

$\forall t \in [t_{i,k}^c, t_{i,k}^u) \cap [\tau_y, \infty) \cap [\tau_i, \infty) \quad \forall k \in \mathbb{N}$, where $\lambda_{2,i} \triangleq \frac{1}{\beta_2}\min\{k_{1,i}\underline{\alpha}_i, k_y, k_{\text{CL},y}\underline{\lambda}_y, k_{\text{CL},i}\underline{\lambda}_i\}$. Applying the comparison lemma [40, Lemma 3.4] to (39) yields

$$V_i(z_i(t)) \leq V_i\left(z_i\left(t_{i,k}^c\right)\right)\mathrm{e}^{-\lambda_{2,i}\left(t-t_{i,k}^c\right)} + \kappa_{2,i} \tag{40}$$

$\forall t \in [t_{i,k}^c, t_{i,k}^u) \cap [\tau_y, \infty) \cap [\tau_i, \infty) \quad \forall k \in \mathbb{N}$, where $\kappa_{2,i} \triangleq \frac{C_{1,i}}{\lambda_{2,i}}$. Then, (40) can be used with (28) to yield (34). ■

B. Target Operating in the Unchased Mode

In this section, the states associated with the ith target are shown to be bounded during periods when it is not the currently *chased*target. Here, the subscript $i = c$ refers to the target that is currently operating in the *chased* mode. These terms appear due to the fact that the virtual and actual control laws in (10) and (13), respectively, are designed in terms of the currently *chased* target and, thus, will appear in this analysis with the inclusion of the state e_y in (27).

Lemma 2: During $t \in [t_{i,k}^u, t_{i,k+1}^c) \quad \forall k \in \mathbb{N}$, the system states associated with the ith target remain bounded.

Proof: Provided that the sufficient gain conditions in (31)–(33) are satisfied, and substituting (2), (14), (21), and (24), the time derivative of (27) during $t \in [t_{i,k}^u, t_{i,k+1}^c) \quad \forall k \in \mathbb{N}$ can be upper bounded as

$$\begin{aligned}\dot{V}_i \leq & \bar{\alpha}_i \|\bar{x}_i\|^2 + \bar{\alpha}_i \|\bar{x}_i\| \|x_{i,g}\| + \bar{\alpha}_i \|\bar{x}_i\| \|e_y\| \\ & + K_{1,i}\bar{\alpha}_i \|\bar{x}_i\| \|\bar{x}_c\| + \bar{\alpha}_i \|\bar{x}_i\| \left\|x_{i,g}^c\right\| + \|\bar{x}_i\| \|f_i\| \\ & + K_{1,i} \|e_y\| \left\|W_i^T \sigma_i\right\| + K_{1,i} \|e_y\| \left\|\hat{W}_c^T \sigma_c\right\| - k_y \|e_y\|^2.\end{aligned} \tag{41}$$

Using Young's inequality and (28), and upper bounding the term $\|\bar{x}_c\|$ using the right-hand side of (30), (41) can be further upper bounded as

$$\dot{V}_i(z_i(t)) \leq \lambda_{3,i} V_i(z_i(t)) + C_{2,i} \tag{42}$$

where $\lambda_{3,i}, C_{2,i} \in \mathbb{R}$ are positive constants, and $\lambda_{3,i} = \frac{1}{\beta_1}\max\left\{\frac{\bar{\alpha}_i}{2}(5 + K_{1,i}) + \frac{1}{2}, K_{1,i} - k_y\right\}$. Applying the comparison lemma [40, Lemma 3.4] to (42), and upper bounding, yields

$$V_i(z_i(t)) \leq \left(V_i\left(z_i\left(t_{i,k}^u\right)\right) + \kappa_{3,i}\right)\mathrm{e}^{\lambda_{3,i}\left(t-t_{i,k}^u\right)} - \kappa_{3,i} \tag{43}$$

where $\kappa_{3,i} \triangleq \frac{C_{2,i}}{\lambda_{3,i}}$. Furthermore, $\forall t \in [t_{i,k}^c, t_{i,k}^u) \cap [\tau_y, \infty) \cap [\tau_i, \infty) \quad \forall k \in \mathbb{N}$ (i.e., once enough data have been collected), the term $C_{2,i}$ in (42) is replaced with a smaller constant $C_{3.i}$ [$C_{3,i} < C_{2,i}$ because (34) is used to upper bound the term $\|\bar{x}_c\|$ instead of (30), and $\kappa_{2,i} < \kappa_{1,i}$], and the resulting differential inequality yields the solution

$$V_i(z_i(t)) \leq \left(V_i\left(z_i\left(t_{i,k}^u\right)\right) + \kappa_{4,i}\right)\mathrm{e}^{\lambda_{3,i}\left(t-t_{i,k}^u\right)} - \kappa_{4,i} \tag{44}$$

where $\kappa_{4,i} = \frac{C_{3,i}}{\lambda_{3,i}}$. ■

C. Combined Analysis

In this section, switched systems analysis is used to show that the states associated with the ith target converge to an ultimate bound. First, (38) and (43) are used to develop an ultimate bound during the learning phase (i.e., before enough data have been collected), and then, an ultimate bound on the states during the second phase (i.e., once learning has occurred) is synthesized based on (40) and (44). To facilitate the analysis, let $\nu_{1,i}, \nu_{2,i}$ denote positive constants $\forall i \in \{1, 2, ..., n_t\}$, where $\nu_{1,i} \triangleq \mathrm{e}^{\lambda_{3,i} T_i^u(km,(k+1)m-1) - \lambda_{1,i} T_i^c(km,(k+1)m-1)}$, $m \in \mathbb{N}$, $\lambda_{1,i}$ is introduced in (37), and $\lambda_{3,i}$ is introduced in (42).

Theorem 1: The controllers in (10) and (13) and the adaptive update laws in (19) and (20) ensure that all signals associated with the ith target remain bounded for all time $t \in [0, \bar{\tau}_i)$, where $\bar{\tau}_i \triangleq \max\{\tau_y, \tau_i\}$, and

$$\limsup_t \|z_i(t)\|^2 \leq \frac{\nu_{2,i}}{\beta_1(1-\nu_{1,i})} \mathrm{e}^{\lambda_{3,i} T_{i,\max}^u} \tag{45}$$

where $T_{i,\max}^u \triangleq \sup_k T_i^u(km, (k+1)m-1)$, provided there exists an $m < \infty$ and sequences $\{\Delta t_{i,k}^c\}_{k=0}^{\infty}$ and $\{\Delta t_{i,k}^u\}_{k=0}^{\infty}$ such that $\forall k \in \mathbb{N}$, the sufficient condition

$$T_i^u(km, (k+1)m-1) < \frac{\lambda_{1,i}}{\lambda_{3,i}} T_i^c(km, (k+1)m-1) \tag{46}$$

is satisfied.

Proof: Consider a single cycle of the ith target switching from *chased* to *unchased* and back to the *chased* mode, i.e., $t \in [t_{i,k}^c, t_{i,k+1}^c)$. Using (38) and (43), the evolution of V_i over m cycles can be written as

$$V_i\left(z_i\left(t_{i,(k+1)m}^c\right)\right) \leq \nu_{1,i} V_i\left(z_i\left(t_{i,km}^c\right)\right) + \nu_{2,i}$$

where $\nu_{1,i} < 1$ provided (46) is satisfied. Let $\{s_{i,k}\}_{k=0}^{\infty}$ be a sequence defined by the recurrence relation $s_{i,k+1} = M_{1,i}(s_{i,k})$, with initial condition $s_{i,0} = V_i(z_i(t_{i,0}^c))$, where $M_{1,i} : \mathbb{R} \to \mathbb{R}$ is defined as $M_{1,i}(s) \triangleq \nu_{1,i} s + \nu_{2,i}$. Since $\nu_{1,i} < 1$, $M_{1,i}$ is a contraction [41, Definition 9.22], and thus all initial conditions $s_{i,0}$ approach the fixed point $s = \frac{\nu_{2,i}}{1-\nu_{1,i}}$ [41, Th. 9.23]. Since the sequence $\{s_{i,k}\}$ upper bounds V_i, in the sense that $V_i(z_i(t_{i,km}^c)) \leq s_{i,k}$, V_i is ultimately bounded. However, since the dwell-time condition in (46) is specified over m cycles rather than a single cycle, V_i may grow within $[t_{i,km}^c, t_{i,(k+1)m}^c]$. Thus, the ultimate bound of z_i is given by (45). ■

The analysis in Theorem 1 shows that the closed-loop system is ultimately bounded during the initial phase (i.e., $t \in [0, \bar{\tau}_i)$). To further reduce the ultimate bound on the states associated with the ith target, the following theorem uses (40) and (44) to establish a smaller bound once sufficient excitation occurs (i.e., $t \in [\bar{\tau}_i, \infty)$). To facilitate the following theorem, let $\nu_{3,i}, \nu_{4,i}$ denote positive constants, where $\nu_{3,i} \triangleq \mathrm{e}^{\lambda_{3,i} T_i^u(km,(k+1)m-1) - \lambda_{2,i} T_i^c(km,(k+1)m-1)}$, $m \in \mathbb{N}$, $\lambda_{2,i}$ is introduced in (39), and $\lambda_{3,i}$ is introduced in (42).

Theorem 2: The controllers in (10) and (13), and the adaptive update laws in (19) and (20) ensure that all signals associated with the ith target remain bounded for all time $t \in [\bar{\tau}_i, \infty)$ and

$$\limsup_t \|z_i(t)\|^2 \leq \frac{\nu_{4,i}}{\beta_1(1-\nu_{3,i})} \mathrm{e}^{\lambda_{3,i} T_{i,\max}^u} \tag{47}$$

provided there exists an $m < \infty$ and sequences $\{\Delta t_{i,k}^c\}_{k=0}^{\infty}$ and $\{\Delta t_{i,k}^u\}_{k=0}^{\infty}$ such that $\forall k \in \mathbb{N}$, the sufficient condition

$$T_i^u(km, (k+1)m-1) < \frac{\lambda_{2,i}}{\lambda_{3,i}} T_i^c(km, (k+1)m-1) \tag{48}$$

is satisfied.

Proof: This proof follows the same strategy as that of Theorem 1 for $t \in [t_{i,k}^c, t_{i,k+1}^c) \cap [\bar{\tau}_i, \infty)$. Provided (48) is satisfied $\nu_{3,i} < 1$. By establishing $\{s_{i,k}\}_{k=0}^{\infty}$ as a sequence defined by the recurrence relation $s_{i,k+1} = M_{2,i}(s_{i,k})$ with initial condition $s_{i,0} = V_i(z_i(t_{i,q_i}^c))$, where $q_i \triangleq \mathrm{argmin}_k\{t_{i,k}^c > \bar{\tau}_i\}$ and $M_{2,i} : \mathbb{R} \to \mathbb{R}$ is defined as $M_{2,i}(s) \triangleq \nu_{3,i} s + \nu_{4,i}$, then following the same steps as in Theorem 1, the result in (47) can be concluded. ■

Remark 2: The main result[11] of Theorems 1 and 2 [i.e., (45) and (47), respectively] can be compared to a uniformly ultimately bounded (UUB) result in a control system, but with the inability to explicitly define the transient state behavior (analogous to an asymptotic stability result) for the overall trajectories. However, as is apparent in (38) and (43) for Theorem 1, and in (40) and (44) for Theorem 2, a bound on the transient behavior is defined in the stable and unstable subsystems (i.e., during each activation of the *chased* and *unchased* modes, respectively). So while this interesting information is lost in the final result for an arbitrary switching strategy, the overall transient behavior actually could be defined for a predetermined switching signal (where it is preplanned exactly when to switch and to which target to switch), although that type of strategy makes little sense for this herding application. This challenge arises from the existence of the constant terms in (38), (40), (43), and (44), as they make the expressions incompatible with traditional switched systems analysis tools, motivating the use of the method in this paper.

Remark 3: Let $\bar{T}_{\text{tot}}^c \in \mathbb{R}$ and $\bar{T}_{\text{tot}}^u \in \mathbb{R}$ denote the average total time target agents spend operating in the *chased* and *unchased* modes, respectively. Using (46) and (48), an average dwell-time condition for all target agents over all time can be written as

$$\bar{T}_{\text{tot}}^u < \frac{\lambda_{\text{chased}}}{\lambda_{\text{unchased}}} \bar{T}_{\text{tot}}^c \tag{49}$$

where $\lambda_{\text{chased}} \triangleq \min_{i \in \mathcal{T}}\{\lambda_{1,i}, \lambda_{2,i}\}$ and $\lambda_{\text{unchased}} \triangleq \max_{i \in \mathcal{T}}\{\lambda_{3,i}\}$. Since only one target will operate in the *chased* mode at any given time, for n_t targets, the average total time targets spend operating in the *chased* mode is $\bar{T}_{\text{tot}}^c \triangleq \frac{1}{n_t}\left(\bar{T}_{\text{tot}}^c + \bar{T}_{\text{tot}}^u\right)$. From (49), a sufficient condition for the maximum number of target agents that a single herding agent can successfully regulate is $n_t < \frac{\lambda_{\text{chased}}}{\lambda_{\text{unchased}}} + 1$.

[11] This result can be interpreted to state that as the system approaches steady state (as time goes to infinity), the error signals associated with the ith target may continue to grow and decay, but that the errors will be bounded by an expression that is based on constants that can be manipulated to meet the desired objective by tuning the gain selections and switching strategy.

Algorithm 1: Switching Strategy.

while not all targets lie within the neighborhood of their goal location **do**

1. herder selects the target furthest from the goal location to be the *chased* target
2. chase target until $\|\bar{x}_i^c(t)\| \leq \gamma_r \left\|\bar{x}_i^c\left(t_{i,k-1}^u\right)\right\|$
3. break

end while

VI. Results

To demonstrate the developed control strategy, the following results are presented. First, simulations show the efficacy of this control strategy when the herder is challenged with multiple targets, fleeing both the herder and their goal locations, and results are provided showing that the herding objective is accomplished and the function approximation scheme is effective. Then, experimental results are given, demonstrating the ability to apply this strategy to a physical system including an autonomous quadcopter herding agent, and six unactuated target agents that respond to the quadcopter in an unknown way. Finally, the results of a parameter survey are presented, providing data for interesting discussion on how to design the switching strategy.

A. Implementation and Switching Strategy

Besides the need to model the unknown functions in (1) and (2) in simulated trials, only the measurable states, desired goal locations, and selected parameters (gains, initial conditions, switching strategy, etc.) are required for implementation of this control strategy. There are numerous computed parameters introduced throughout this paper, which are only included for informational purposes in the analysis. Each of the gain conditions, dwell-time conditions, and the number of targets condition are sufficient, not necessary, conditions, and while informative, are not required to implement this method.

For each of the following demonstrations, an agent is considered to be "herded" once it reaches a distance within the tolerance, $R \in \mathbb{R}_{>0}$, of their goal location. Once a target is herded, it stops moving and is no longer considered in the switching strategy. The control development and stability analysis is agnostic to the specific design of the herder's switching strategy, except for sufficient dwell-time conditions that must be satisfied to ensure stability. The herder switching strategy used in the following simulations and experiments is defined by Algorithm 1, which depends on the herding ratio $\gamma_r \in (0,1)$ and is selected such that the sufficient dwell-time conditions [(46) for $t \in [0, \bar{\tau}_i))$, and (48) during $t \in [\bar{\tau}_i, \infty))$] are met. The analysis in Section V is completely agnostic to the design of the switching strategy and, therefore, does not directly dictate the selection of the herding ratio. In other words, the herding ratio is a design parameter that is selected such that the dwell-time conditions are satisfied and the controller performance is acceptable.

B. Simulations

The following simulations are included to show how the developed strategy performs when tasked with herding targets that have additional dynamics other than just fleeing the herder, specifically that they also seek to avoid their prescribed goal location. In addition, the simulations demonstrate the ability of the function approximation strategy to estimate the true dynamics. The machine used for all simulation trials featured an Intel Core i5-2500 processor with 8GB DDR3 RAM and a 250GB SSD.

For each simulation, the unknown functions in (2) were selected to be

$$\alpha_i\left(\|x_i - y\|\right) = a_i \exp\left(-\frac{1}{2\sigma}\left(x_i - y\right)^T\left(x_i - y\right)\right)$$

and

$$f_i\left(x, y\right) = b_i \bar{x}_i \exp\left(-\frac{1}{2\sigma}\left(x_i - y\right)^T\left(x_i - y\right)\right)$$

where $a_i \in \mathbb{R}_{>0}$, $b_i \in \mathbb{R}_{>0}$, and $\sigma \in \mathbb{R}_{>0}$ are unknown constants, whereas the unknown function in (1) was chosen as $h(x, y) = 0$, for simplicity. This model, taken from [9], dictates that the targets are each repelled from the herder, while also fleeing their goal location, with both behaviors scaled by their distance to the herder. In other words, the targets are each inclined to remain relatively stationary (similar to grazing) unless the herder is nearby, in this case, they flee the herder and their goal location. The function approximation scheme given in Section IV-B was used to learn and compensate for the unknown dynamics in real time. For each of the three simulations, the overall herder and target trajectories will be shown, as well as a less cluttered plot showing just the starting and ending positions of each agent. In addition, the target herding error norms and target dynamics function approximation percent error $\eta_i \in \mathbb{R}_{\geq 0}$ are given for each simulation.

1) Simulation 1. All Agents to Origin: This simulation tasked a single herder with the objective of regulating eight target agents from their initial positions to a common neighborhood about the origin. The parameters were selected as $a = [6.2 \quad 7.3 \quad 6.7 \quad 7.6 \quad 7.4 \quad 6.9 \quad 7.1 \quad 7.7]$, $b = [0.68 \quad 0.71 \quad 0.62 \quad 0.77 \quad 0.73 \quad 0.69 \quad 0.63 \quad 0.61]$, $\sigma = 2$, $\gamma_r = 0.65$, and the goal neighborhood radius $R = 2.5$ m. In Fig. 1, the overall paths of all agents are shown, whereas just the starting and ending positions are shown in Fig. 2. Fig. 3 shows the target error norms and Fig. 4 shows the plot of the target velocity function approximation percent error for this simulation.

2) Simulation 2. Agent Formation: The second simulation also features one herder and eight targets, with the same dynamics parameters as in Simulation 1, but with the herding objective dictating that each target be regulated to different locations in the workspace, resulting in a predefined target agent formation. For this simulation, the herding ratio was selected to be $\gamma_r = 0.45$, and the goal neighborhood size was $R = 2.0$ m. The overall paths of all agents are shown in Fig. 5, whereas the starting and ending positions are shown alone in Fig. 6.

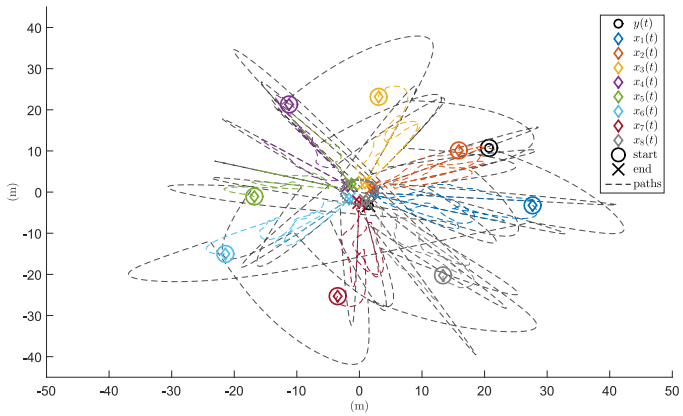

Fig. 1. Simulation 1: Overall trajectories for all agents are shown.

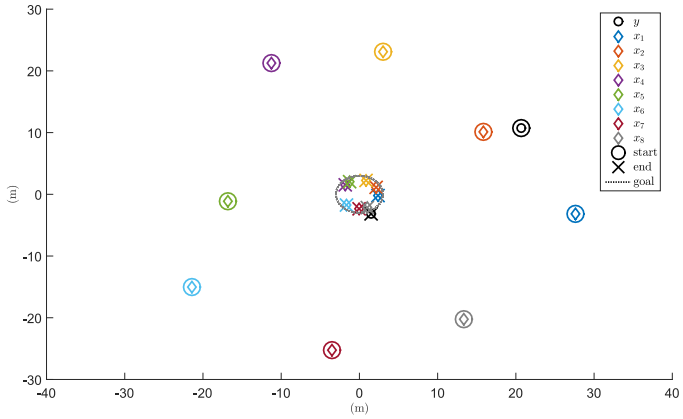

Fig. 2. Simulation 1: This figure shows the starting and ending positions of each agent to clearly show that the herding objective is accomplished.

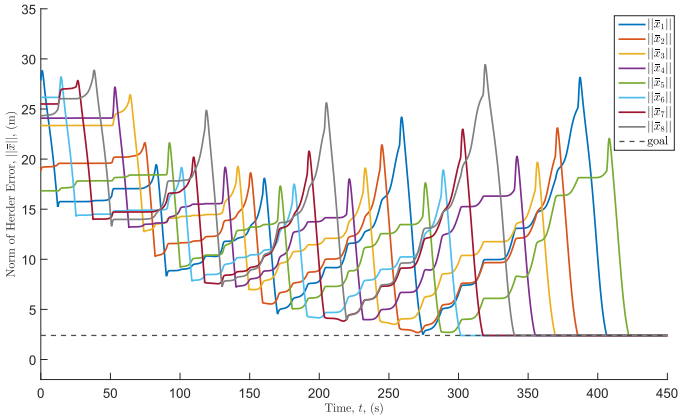

Fig. 3. Simulation 1: The target error norms are shown.

Figs. 7 and 8 show the target error norms and the function approximation percent error, respectively.

3) Simulation 3. Formation in Three Dimensions (3-D): The third simulation demonstrates the ability of herding in 3-D space, with one herder and four targets. To define the dynamics of the targets, the parameters were selected to be $a = [10.2 \quad 11.3 \quad 10.7 \quad 11.6]$, $b = [3.0 \quad 4.2 \quad 3.8 \quad 4.5]$,

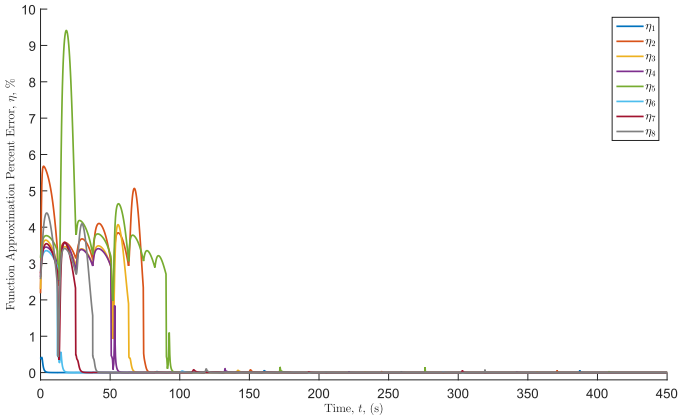

Fig. 4. Simulation 1: Function approximation percent error for the target dynamics is shown.

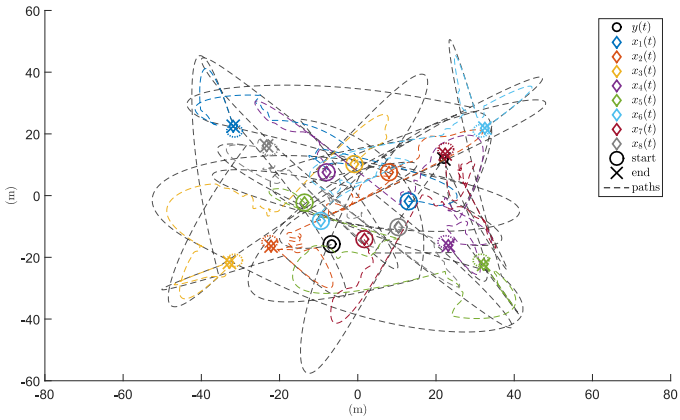

Fig. 5. Simulation 2: Trajectories of each agent are shown.

$\sigma = 2.5$. The herder in this simulation is tasked with regulating the target agents to a tetrahedron-shaped formation near the origin, as seen in the ending configuration shown in Fig. 9(a). The herding ratio was selected to be $\gamma_r = 0.35$, and the goal neighborhood size was $R = 1$ m. The three-dimensional paths of all agents are shown in Fig. 9(b). The target error norms are given in Fig. 10, whereas the function approximation percent error is shown in Fig. 11.

C. Experiments

Experiments[12] were performed to illustrate the performance of the developed controller. First, two experiments with different objectives were conducted using the Parrot Bebop 2 quadcopter and unactuated paper targets in Fig. 12 to demonstrate the efficacy of the proposed approach. Then, a series of addition trials were performed to study the effect that the herding ratio parameter has on performance. The Parrot Bebop 2 quadcopter platform served as the herding agent while the targets were mobile platforms constructed from paper plates and poster board, designed to slide on the lab floor without tipping over. Each target was

[12]A video of a typical run of Experiments 1 and 2 is available at [42].

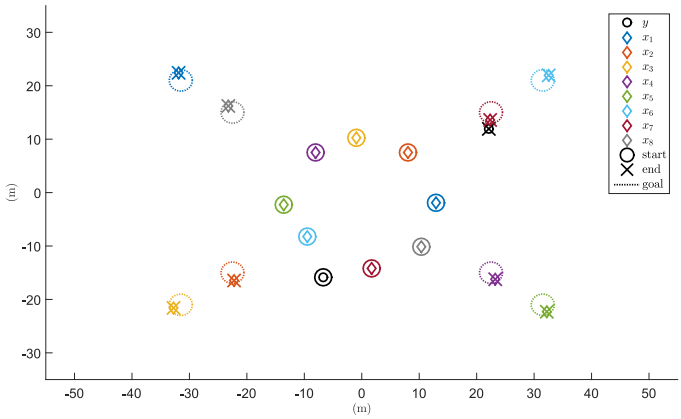

Fig. 6. Simulation 2: This figure shows the starting and ending positions of the herder and each target agent. After starting in a roughly 20 m diameter circle formation around the origin, the targets final configuration resembles a wide "X" pattern.

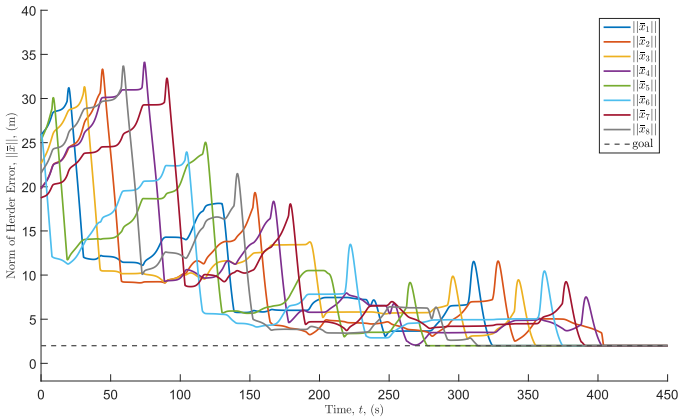

Fig. 7. Simulation 2: Target error norms are shown, with each being regulated to the 2 m tolerance.

constructed from a paper plate on the top and bottom, fastened to varying heights of colored poster board, with arbitrarily sized pieces of cardboard glued inside, to create cylinders of different sizes and weights. Although the herder hovers above targets located on the ground plane, these experiments were implemented as though all agents were operating on the same plane (i.e., the herder was restricted to a particular altitude and executed the designed controller in two-dimensional space). The curved shape of the top edges of the target agent platforms allow the column of air generated by the quadcopter propellers to cause nearby target agents to slide away from the herder. In each experiment, the herder (quadcopter) was tasked with relocating six targets to specific locations without knowledge of the herder–target interaction (i.e., the dynamics of the paper targets sliding on the floor and the aerodynamic forces from the quadcopter acting on the paper targets are unknown).

The centralized machine used for all computations and responsible for sending velocity commands to the herder agent featured an Intel Core i7-8700 processor with 32GB DDR4 RAM and a 250GB SSD. A NaturalPoint, Inc. OptiTrack

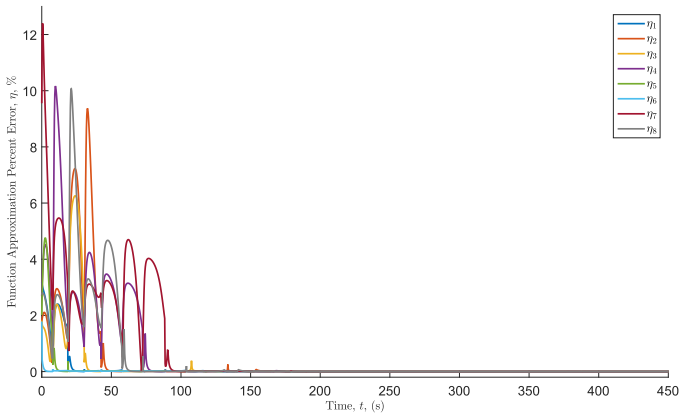

Fig. 8. Simulation 2: Target dynamics function approximation percent error is shown.

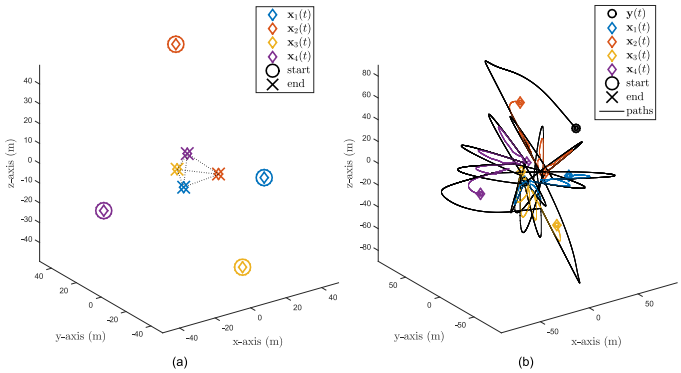

Fig. 9. Simulation 3: (a) Starting and ending positions of each target agent are shown. The dotted lines connecting the agents are included to help display the 3-D formation of the ending configuration. (b) 3-D paths of each agent are shown, with each agents path shown as a solid line.

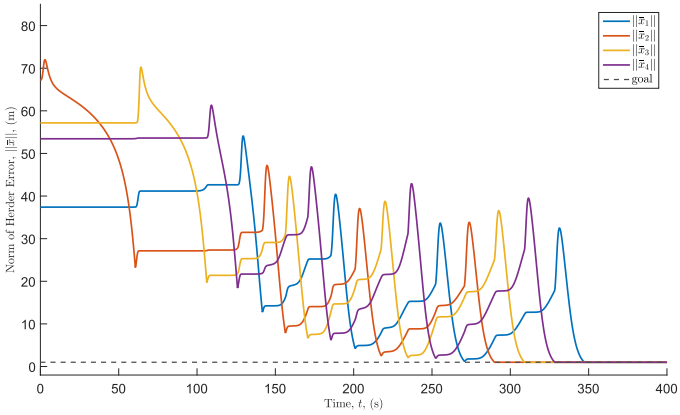

Fig. 10. Simulation 3: Norms of the target errors are shown. Each target agent is regulated to the desired threshold of 1 m.

motion capture system was used to record the position of each agent at all times for feedback control at a rate of 120 Hz.

1) Experiment 1. All Agents to Origin: For Experiment 1, the herder was tasked with regulating a set of six randomly dispersed agents to a neighborhood about the origin. The herding ratio was

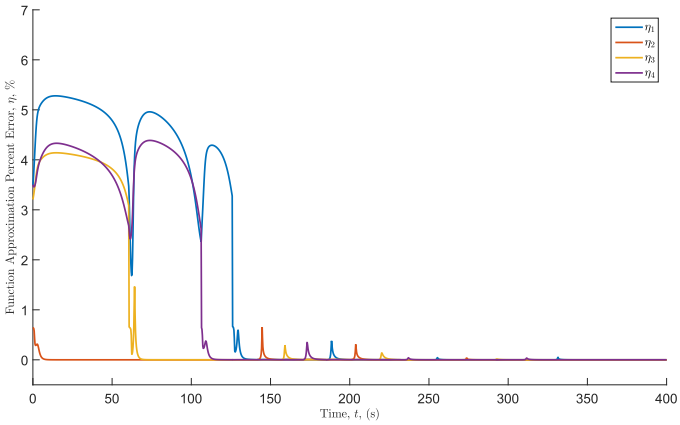

Fig. 11. Simulation 3: This figure shows the function approximation percent error for the 3-D simulation.

Fig. 12. Parrot Bebop 2 quadcopter used for the herder as well as the paper platforms that represent the targets is shown. The markers seen on each agent are infrared reflectors arranged in unique configurations, which are used with the NaturalPoint, Inc. OptiTrack motion capture system to track the positions of each agent for feedback control at 120 Hz. Each target agent platform has a different height and weight, and are designed to slide without tipping on the floor shown.

selected to be $\gamma_r = 0.25$, and the goal neighborhood size was $R = 0.6$ m. Fig. 13 shows the overall paths and starting and ending positions of all agents. Fig. 14 depicts the norms of each target error.

2) Experiment 2. Separate Agents by Color: The objective for the Experiment 2 was to regulate the target agents to different locations based on their color (the color of each agent is known). The herding ratio and goal neighborhood size were selected to be $\gamma_r = 0.6$, and $R = 0.5$ m, respectively. Fig. 15 depicts the starting and ending positions of the pink and yellow targets, and the path that each agent traversed. Target error norms for Experiment 2 are depicted in Fig. 16.

3) Experiment 3. Method Comparison: The setup for this group of experiments is identical to that in Experiment 1, for the purpose of comparing the method in this paper to the SMC

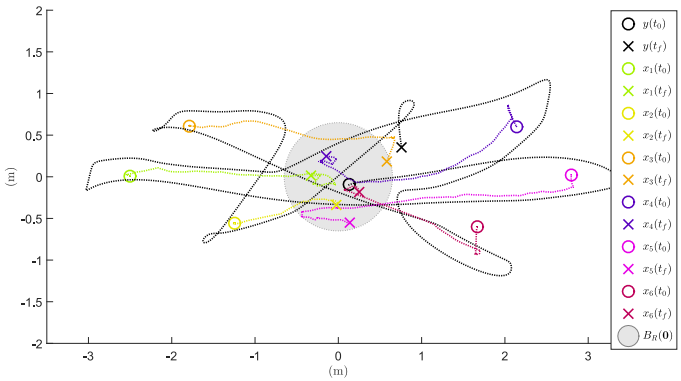

Fig. 13. Experiment 1: Herder and target paths from initial locations denoted by circles and final locations denoted by X's, where the dotted lines represent the path of each agent. The final location of all target agents are regulated within the goal radius denoted by the light gray circular region.

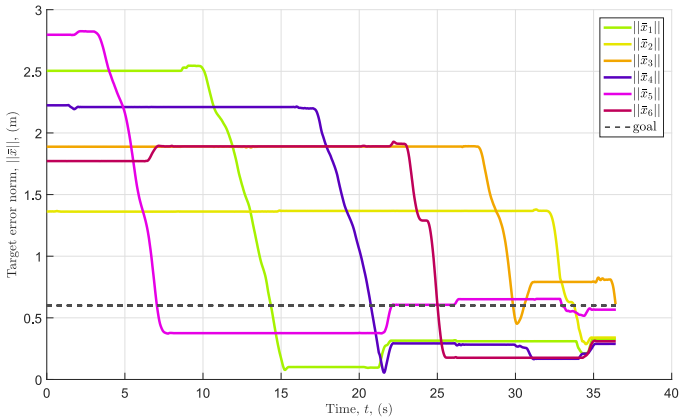

Fig. 14. Experiment 1: Shown here is the norm of the target herding errors. Each target agent is regulated within the desired threshold of 0.6 m.

in [29]. For two different values of the herding ratio γ_r, the average target error is plotted for the compared methods. Fig. 17 shows the average target errors for $\gamma_r = 0.5$ and Fig. 18 shows the same for $\gamma_r = 0.8$. Furthermore, Table I shows the average root-mean-square (rms) error ($\bar{x}_{\text{aveRMS}}$) and average time-to-completion (TTC) for each controller, demonstrating that this method slightly out performs the SMC in [29].

4) Experimental Parameter Survey: Using the same experimental setup as in Experiments 1 and 2, this section details the results of several trials with different values for the herding ratio. Table II compares the average rms error (p-value $= 0.025$) and TTC (p-value $= 0.003$) for multiple different herding ratio values for the Experiment 1 objective. Similarly, Table III displays the results of the average rms error (p-value $= 0.001$) and TTC (p-value $= 0.011$) for multiple different herding ratio values for the Experiment 2 objective.

VII. Discussion

A. Simulations Discussion

The unknown interactions between the herder and targets were approximated using an NN for each target. Each NN consisted of $L_1 = 200$ basis functions, with centers randomly

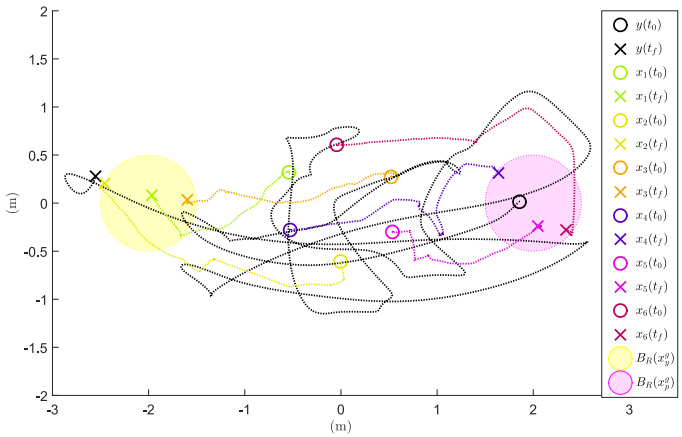

Fig. 15. Experiment 2: The herder and target paths from initial locations denoted by circles to final locations denoted by X's, where the dotted lines represent the path of each agent. The final location of all target agents are regulated within the goal radius denoted by the yellow or pink shaded circular regions.

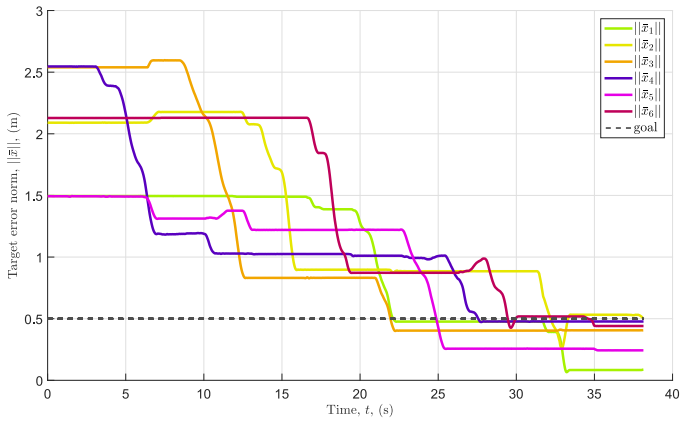

Fig. 16. Experiment 2: Norm of the target herding errors are shown, each regulated within the desired threshold of 0.5 m.

initialized throughout four 4×4 m quadrants around the herder, and standard deviations of 0.5 m. During the simulations, the function approximation error for each target would decay during the periods that target was selected as the *chased* target, and would remain stationary when in the *unchased* mode. As seen in Figs. 4, 8, and 11, sometimes there would be a small spike in approximation error when a target was switched back into the *chased* mode, but it would promptly decay again, demonstrating the effectiveness of the proposed function approximation strategy.

For Simulation 1, where the objective entailed regulating all targets near the origin, the target error norms would actually grow slightly over time, as seen in Fig. 3. This was caused by the fact that the defined model dictated that the targets' desire to flee the herder and goal location is scaled by that target's proximity to the herder. So when all targets are being sent to the same location, they are more likely to be in close proximity to the herder, and thus more aggressively move away from the herder and goal. This led to the active behavior within about 30 m of the origin seen in Fig. 1, where the herder being so close to the targets caused increased fleeing speeds.

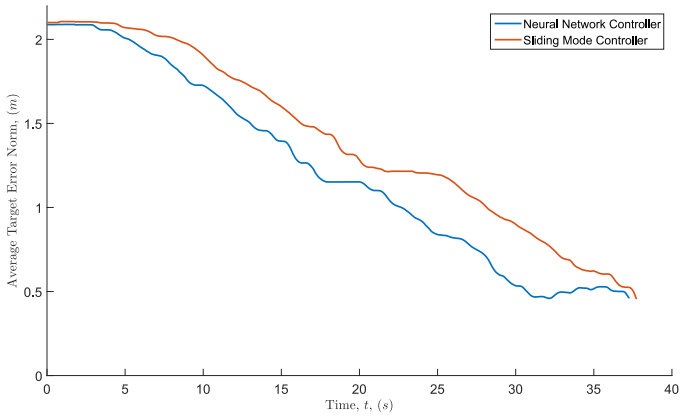

Fig. 17. Experiment 3: The average norm of the target errors for $\gamma_r = 0.5$. The SMC from [29] was slightly outperformed by the method in this paper.

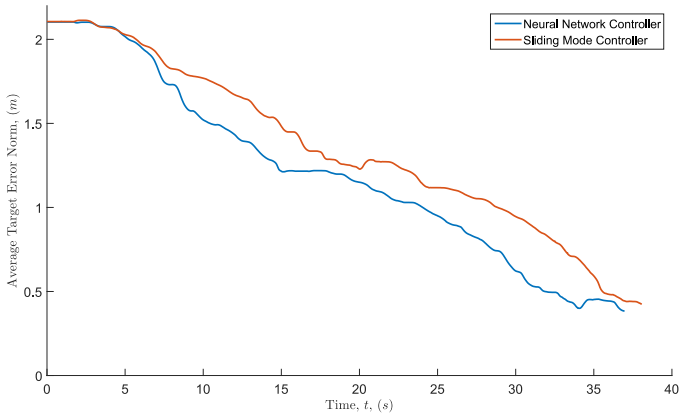

Fig. 18. Experiment 3: Average norm of the target errors for $\gamma_r = 0.8$. This method performed slightly better than the SMC from [29].

TABLE I
CONTROL METHOD COMPARISON

γ_r	Controller	$\bar{x}_{\text{aveRMS}}$, (m)	TTC, (s)
0.5	NN	1.512	36.83
	SMC	1.581	39.89
0.8	NN	1.438	37.53
	SMC	1.491	39.18

In contrast, the Simulation 2 objective involved all targets being regulated to unique locations in the workspace, resulting in a final target formation. In this case, Fig. 7 shows the target error norms growing slightly at first, before being quickly driven down. Once the targets are mostly separated, the targets become less and less likely to be influenced by the herder inadvertently, and each target can be methodically driven to its goal location. It is quite interesting that, while certain herding ratios yielded better results than others, the distinct behaviors between the two objectives described earlier would always be somewhat evident.

The results from Simulation 3 exhibited behaviors similar to Simulation 2, while also demonstrating that the method in this paper is not limited to a two-dimensional (2-D) plane. With less targets, higher $\frac{a}{b}$ ratios, and a smaller goal tolerance in this case,

TABLE II
PARAMETER SURVEY 1

γ_r	$\bar{x}_{\text{aveRMS}}$, (m)	TTC, (s)
0.2	1.559	39.17
0.4	1.503	38.87
0.5	1.512	36.83
0.6	1.459	36.94
0.8	1.438	44.71

TABLE III
PARAMETER SURVEY 2

γ_r	$\bar{x}_{\text{aveRMS}}$, (m)	TTC, (s)
0.2	1.375	38.56
0.4	1.080	31.17
0.5	1.143	33.18
0.6	1.278	36.29
0.8	1.079	36.30

it can be seen in Fig. 10 that the targets very nearly reach the goal a few times before finally arriving.

B. Experiments Discussion

The NN for each target agent used $L_1 = 200$ basis functions, with centers randomly initialized throughout four 2×2 m quadrants around the herder, and standard deviations of 0.3 m. The herder estimated the repulsion effects that it exhibited on each target in the quadcopter body frame, meaning that target velocity vectors pointing in the forward direction of the quadcopter had positive magnitude, and those pointing in the opposite direction were negative.

The learning scheme, considering only the magnitudes of the interactions based on herder–target relative positions, was unable to identify other apparent interactions, such as those that are velocity dependent. For instance, the effect that the herder may have on a target 1 m away while stationary may be drastically different from the effect it has on a target 1 m away while moving. Additionally, forces due to friction between the paper platform and the floor will vary depending on the velocity of the platform.

For the two different experiment setups, the average rms error and the time to complete a successful run varied in interesting ways depending on the value of the herding ratio γ_r. Based on the results in Tables II and III, the best values for γ_r were those closer to 0.5, which provides a good balance of minimizing long travel time between switches while still being able to separate clustered targets. For the first objective, higher values of γ_r yielded slightly better average rms errors, but the highest average TTC resulted from the highest herding ratio, with the perceived "sweet spot" being around $\gamma_r = 0.6$. Similarly for the second objective, the best average TTC values were seen with the middle values of γ_r, but the rms errors were less informative in this case. This may be in part due to the fact that many more random target collisions and obstructions occur in the early parts of these trials, as all agents are initially placed relatively close together. The best trials for the second objective used the herding ratio $\gamma_r = 0.4$.

The method in this paper was also compared to the method that solves the one versus many indirect herding problem in [29]. As seen in Figs. 17 and 18, this method yields better average errors than the SMC method, which is also confirmed by the results of the trials shown in Table I.

VIII. CONCLUSION

This paper considered the challenging indirect herding problem, where a single controlled agent is responsible for regulating the position of a numerous indirectly controlled agents to desired set points. A unique constraint to this problem was the fact that the herder was outnumbered by the target agents, which motivated the need for a switched systems analysis and the development of a dwell-time condition that informe the herder of the minimum time that it must pursue a particular agent before it pursues a different agent. To account for the challenge that the motion of each target agent can only be indirectly controlled through an unknown distance-based herder–target influence function, NNs were used along with a concurrent learning strategy that facilitated the dwell-time condition development. A herder control law and sufficient dwell-time conditions were developed using Lyapunov-based stability analysis to ensure UUB regulation of n_t uncertain heterogeneous target agents to their goal locations. Simulations and experiments were performed to demonstrate the performance of the theoretical results for an example herding strategy. An interesting extension to this paper is a case with multiple herders along with some form of coordination strategy. For example, the results in [27] and [28] provided a foundation for such considerations. Additionally, in scenarios where vision sensing is required to localize targets, herding objectives may still be accomplished by leveraging techniques from [43] to ensure reliable feedback. Other considerations could include limitations of the herding agent (i.e., control saturation, mobility constraints), velocity-dependent herder–target interactions, offline NN training to improve performance, and dealing with additional challenges associated with different switching strategies.

REFERENCES

[1] M. Egerstedt and X. Hu, "Formation constrained multi-agent control," *IEEE Trans. Robot. Autom.*, vol. 17, no. 6, pp. 947–951, Dec. 2001.
[2] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Flocking in fixed and switching networks," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 863–868, May 2007.
[3] M. Ji and M. Egerstedt, "Distributed coordination control of multiagent systems while preserving connectedness," *IEEE Trans. Robot.*, vol. 23, no. 4, pp. 693–703, Aug. 2007.
[4] H. Wang, "Flocking of networked uncertain Euler-Lagrange systems on directed graphs," *Automatica*, vol. 49, no. 9, pp. 2774–2779, 2013.
[5] G. Wen, Z. Duan, Z. Li, and G. Chen, "Flocking of multi-agent dynamical systems with intermittent nonlinear velocity measurements," *Int. J. Robust. Nonlinear Control*, vol. 22, no. 16, pp. 1790–1805, Nov. 2012.
[6] M. Zavlanos, A. Jadbabaie, and G. Pappas, "Flocking while preserving network connectivity," in *Proc. IEEE Conf. Decis. Control*, 2007, pp. 2919–2924.
[7] M. Aranda, G. López-Nicolás, C. Sagüés, and Y. Mezouar, "Formation control of mobile robots using multiple aerial cameras," *IEEE Trans. Robot.*, vol. 31, no. 4, pp. 1064–1071, Aug. 2015.
[8] R. Murray, "Recent research in cooperative control of multivehicle systems," *J. Dyn. Syst. Meas. Control*, vol. 129, pp. 571–583, 2007.

[9] R. Licitra, Z. I. Bell, E. Doucette, and W. E. Dixon, "Single agent indirect herding of multiple targets: A switched adaptive control approach," *IEEE Control Syst. Lett.*, vol. 2, no. 1, pp. 127–132, Jan. 2018.
[10] T. H. Chung, G. A. Hollinger, and V. Isler, "Search and pursuit-evasion in mobile robotics," *Auton. Robots*, vol. 31, no. 4, pp. 299–316, 2011.
[11] R. Vidal, O. Shakernia, H. Kim, D. Shim, and S. Sastry, "Probabilistic pursuit-evasion games: Theory, implementation, and experimental evaluation," *IEEE Trans. Robot. Autom.*, vol. 18, no. 5, pp. 662–669, Oct. 2002.
[12] S. Bhattacharya, S. Hutchinson, and T. Basar, "Game-theoretic analysis of a visibility based pursuit-evasion game in the presence of obstacles," in *Proc. Amer. Control Conf.*, 2009, pp. 373–378.
[13] Y. Ho, A. Bryson, and S. Baron, "Differential games and optimal pursuit-evasion strategies," *IEEE Trans. Autom. Control*, vol. AC-10, no. 4, pp. 385–389, Oct. 1965.
[14] S. D. Bopardikar, F. Bullo, and J. P. Hespanha, "On discrete-time pursuit-evasion games with sensing limitations," *IEEE Trans. Robot.*, vol. 24, no. 6, pp. 1429–1439, Dec. 2008.
[15] V. Isler, S. Kannan, and S. Khanna, "Randomized pursuit-evasion in a polygonal environment," *IEEE Trans. Robot.*, vol. 21, no. 5, pp. 875–884, Oct. 2005.
[16] P. Cheng, "A short survey on pursuit-evasion games," Ph.D. dissertation, Dept. Comput. Sci., Univ. Illinois Urbana-Champaign, Champaign, IL, USA, 2003.
[17] A. D. Khalafi and M. R. Toroghi, "Capture zone in the herding pursuit evasion games," *Appl. Math. Sci.*, vol. 5, no. 39, pp. 1935–1945, 2011.
[18] S. A. Shedied, "Optimal control for a two player dynamic pursuit evasion game: The herding problem," Ph.D. dissertation, Dept. Elect. Comput. Eng., Virginia Polytech. Inst. State Univ., Blacksburg, VA, USA, 2002.
[19] P. Kachroo, S. A. Shedied, J. S. Bay, and H. Vanlandingham, "Dynamic programming solution for a class of pursuit evasion problems: the herding problem," *IEEE Trans. Syst., Man, Cybern.*, vol. 31, no. 1, pp. 35–41, Feb. 2001.
[20] A. S. Gadre, "Learning strategies in multi-agent systems-applications to the herding problem," M.S. thesis, Dept. Elect. Comput. Eng., Virginia Tech, Blacksburg, VA, USA, 2001.
[21] Z. Lu, "Cooperative optimal path planning for herding problems," Ph.D. dissertation, Dept. Aerospace Eng., Texas A&M University, College Station, TX, USA, 2006.
[22] J.-M. Lien, O. B. Bayazit, R. T. Sowell, S. Rodriguez, and N. M. Amato, "Shepherding behaviors," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2004, vol. 4, pp. 4159–4164.
[23] J.-M. Lien, S. Rodriguez, J.-P. Malric, and N. M. Amato, "Shepherding behaviors with multiple shepherds," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 3402–3407.
[24] R. Vaughan, N. Sumpter, J. Henderson, A. Frost, and S. Cameron, "Experiments in automatic flock control," *Robot. Autom. Syst.*, vol. 31, no. 1, pp. 109–117, 2000.
[25] T. Miki and T. Nakamura, "An effective simple shepherding algorithm suitable for implementation to a multi-mobile robot system," in *Proc. 1st Int. Conf. Innov. Comput., Inf., Control*, Washington, DC, USA, 2006, vol. 3, pp. 161–165.
[26] D. Strömbom *et al.*, "Solving the shepherding problem: Heuristics for herding autonomous, interacting agents," *J. Roy. Soc. Interface*, vol. 11, no. 100, 2014, Art. no. 20140719.
[27] M. Bacon and N. Olgac, "Swarm herding using a region holding sliding mode controller," *J. Vib. Control*, vol. 18, no. 7, pp. 1056–1066, 2012.
[28] A. Pierson and M. Schwager, "Controlling noncooperative herds with robotic herders," *IEEE Trans. Robot.*, vol. 34, no. 2, pp. 517–525, Apr. 2018.
[29] R. Licitra, Z. Hutcheson, E. Doucette, and W. E. Dixon, "Single agent herding of n-agents: A switched systems approach," in *Proc. IFAC World Congr.*, 2017, pp. 14374–14379.
[30] A. Parikh, R. Kamalapurkar, and W. E. Dixon, "Integral concurrent learning: Adaptive control with parameter convergence using finite excitation," *Int. J. Adaptive Control Signal Process.*, to be published.
[31] Z. Bell, A. Parikh, J. Nezvadovitz, and W. E. Dixon, "Adaptive control of a surface marine craft with parameter identification using integral concurrent learning," in *Proc. IEEE Conf. Decis. Control*, 2016, pp. 389–394.
[32] M. Krstic, I. Kanellakopoulos, and P. V. Kokotovic, *Nonlinear and Adaptive Control Design*. New York, NY, USA: Wiley, 1995.
[33] F. L. Lewis, "Nonlinear network structures for feedback control," *Asian J. Control*, vol. 1, no. 4, pp. 205–228, 1999.
[34] N. E. Cotter, "The Stone-Weierstrass theorem and its application to neural networks," *IEEE Trans. Neural Netw.*, vol. 1, no. 4, pp. 290–295, Dec. 1990.
[35] W. E. Dixon, A. Behal, D. M. Dawson, and S. Nagarkatti, *Nonlinear Control of Engineering Systems: A Lyapunov-Based Approach*. Boston, MA, USA: Birkhauser, 2003.
[36] G. V. Chowdhary and E. N. Johnson, "Theory and flight-test validation of a concurrent-learning adaptive controller," *J. Guid. Control Dyn.*, vol. 34, no. 2, pp. 592–607, Mar. 2011.
[37] G. Chowdhary and E. Johnson, "A singular value maximizing data recording algorithm for concurrent learning," in *Proc. Amer. Control Conf.*, 2011, pp. 3547–3552.
[38] 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.
[39] G. Chowdhary and E. Johnson, "Concurrent learning for convergence in adaptive control without persistency of excitation," in *Proc. IEEE Conf. Decis. Control*, 2010, pp. 3674–3679.
[40] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.
[41] W. Rudin, *Principles of Mathematical Analysis*. New York, NY, USA: McGraw-Hill, 1976.
[42] R. Licitra and Z. Bell, "Experiments for single agent indirect herding of multiple targets with unknown dynamics," https://youtu.be/8rX5PTao9oI, Apr. 2018. [Online]. Available: https://youtu.be/8rX5PTao9oI
[43] A. Parikh, T.-H. Cheng, H.-Y. Chen, and W. E. Dixon, "A switched systems framework for guaranteed convergence of image-based observers with intermittent measurements," *IEEE Trans. Robot.*, vol. 33, no. 2, pp. 266–280, Apr. 2017.

Ryan A. Licitra received the B.S., M.S., and Doctoral degrees in mechanical engineering from the University of Florida, Gainesville, FL, USA, in 2013, 2015, and 2017, respectively.

He joined the Nonlinear Controls and Robotics group in 2014 to pursue his doctoral research. He is currently with the University of Florida Research and Engineering Education Facility adjacent to Eglin AFB in Ft. Walton Beach, to facilitate extensive collaboration with Air Force Research Laboratory research staff. His research interests include the study of network control systems, including systems with changing topologies, and multitasked agents.

Zachary I. Bell received the bachelor's degree in mechanical engineering in 2015 from the University of Florida, Gainesville, FL, USA, where he is currently working toward the Ph.D. degree in mechanical engineering under the supervision of Dr. W. Dixon.

His research interests include network control, visual estimation, simultaneous localization and mapping, and mechanical design.

Warren E. Dixon (F'16) received the Ph.D. degree in electrical engineering from the Department of Electrical and Computer Engineering, Clemson University, Clemson, SC, USA, in 2000.

He was a Research Staff Member and Eugene P. Wigner Fellow with Oak Ridge National Laboratory until 2004, when he joined the Department Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL, USA. His main research interest has been the development and application of Lyapunov-based control techniques for uncertain nonlinear systems.

Dr. Dixon was the recipient of the 2009 and 2015 American Automatic Control Council O. Hugo Schuck (Best Paper) Award, the 2013 Fred Ellersick Award for Best Overall MILCOM Paper, the 2011 American Society of Mechanical Engineers Dynamics Systems and Control Division Outstanding Young Investigator Award, the 2006 IEEE Robotics and Automation Society Early Academic Career Award, and NSF CAREER Awards during 2006–2011. He is an ASME Fellow.