Decentralized Ergodic Control: Distribution-Driven Sensing and Exploration for Multi-Agent Systems

Ian Abraham and Todd D. Murphey

Abstract— We present ergodic control as a method for providing decentralized time-varying coverage using multiple agents with nonlinear dynamics. Ergodic control allows us to specify distributions as objectives for area coverage problems for nonlinear robot systems. A variant of an ergodic control policy is derived here to enable full distributivity amongst a set of robotic agents. It is then shown that using consensus-based methods, ergodic control can be fully decentralized for a multi-agent system. Examples are presented to illustrate the applicability of our method for multi-agent terrain mapping as well as multi-agent target localization. An analysis on ergodic policies as a Nash equilibrium is provided for game theoretic applications.

I. INTRODUCTION

Within the field of autonomous mobile robots, active exploration and area coverage has been a fundamental area of research. Specifically, exploration and mapping of unknown environments has led to the development of various SLAM-like algorithms [1], [2], [3]. Recent work in active exploration has utilized optimal sensing based on sensor modalities available to the robot [4], [5], [6], [7], [8]. Encoding exploratory behavior in robotics has largely been inspired by biology [9], [10]. In particular, the need for unique exploratory behavior subject to constraints imposed by sensory modalities like visual and tactile has led to advances in the use of information theoretic active exploration [11], [4], [6], [12], [13]. A clear extension to much of the previously mentioned work is application to decentralized multi-agent robotic networks.

Within the field of active exploration and area coverage, distributed robot networks have been shown to improve the efficiency and applicability of exploration of mobile robots [14], [15], [16]. Moreover, the application of distributed robotic networks take into consideration the specifications of the task and effectively use the underlying network connection for improved performance [16], [17], [18], [19].

Existing work in applications of ergodic theory in robotics has been formulated as a centralized multi-agent system for linear dynamical systems [20]. Later work then used a hybrid systems theory approach for ergodic control for nonlinear systems [11] which uses a centralized hub for multi-agent

Authors are with the Neuroscience and Robotics Laboratory (NxR) at the Department of Mechanical Engineering, Northwestern University, 2145 Sheridan Road Evanston, IL 60208 USA.
Email: i-abr@u.northwestern.edu, t-murphey@northwestern.edu

area coverage. In addition, work has been done on synthesizing controllers for multi-robot systems using ergodic processes [21]. However, it has yet to be shown how such a network would benefit from a fully decentralized network or whether it can be extended to nonlinear dynamical systems.

In this work, we contribute a formulation of an ergodic control policy for active exploration and area coverage for multi-agent systems. Specifically, we formulate a control policy for nonlinear dynamical systems which uses hybrid systems theory as in [11], [22]. This formulation is shown to be distributable amongst a multi-agent system. It is then shown that under network consensus in a decentralized multi-agent system, the proposed controller has comparable performance with that of a centralized ergodic control policy. As examples, we apply our algorithm to simulated multi-agent mapping of terrain and target localization problems. Last, using game theory, we analyze ergodic control as a Nash equilibrium control policy for multi-agent games.

The paper is outlined as follows: Section II introduces ergodicity and the ergodic metric as well as formulates the ergodic control problem for decentralized multi-agent systems. Section III demonstrates the algorithm on an area-coverage problem for multi-agents. We then present the problem for multi-agent target localization in Section IV. A game theoretic analysis on ergodic control policies is provided in Section V and the conclusion is in Section VI.

II. DECENTRALIZED ERGODIC CONTROL

In this section, ergodicity and the ergodic metric are introduced. The ergodic metric is then applied in robotics for area coverage with respect to a defined spatial statistic. A variant of an ergodic controller is then derived and shown to be distributable. A consensus-based version of the ergodic metric is then presented to fully decentralize the controller for multi-agent systems.

A. Ergodicity and the Ergodic Metric

Let us define a robot whose state at time t is given as $x(t) : \mathbb{R}^+ \to \mathbb{R}^n$. Controls to the robot at time t are $u(t) : \mathbb{R}^+ \to \mathbb{R}^m$. The dynamics of the robot are assumed to be governed by a control-affine dynamical system of the form

$$\dot{x}(t) = f(x(t), u(t)) = g(x(t)) + h(x(t))u(t) \qquad (1)$$

where $g(x) : \mathbb{R}^n \to \mathbb{R}^n$ is the free, unactuated dynamics of the robot, and $h(x) : \mathbb{R}^n \to \mathbb{R}^{n \times m}$ is the dynamic control response subject to input $u(t)$. Next, define a bounded domain $\mathcal{X}_v \subset \mathbb{R}^v$ whose limits are $[0, L_1] \times [0, L_2] \times \ldots [0, L_v]$ with $v \le n$. A robot's time-averaged statistics $c(s, x(t))$

for a trajectory $x(t)$ (i.e., the statistics describing where the robot spends most of its time) for some time interval $t \in [t_i, t_i + T]$ is given by

$$c(s, x(t)) = \frac{1}{T} \int_{t_i}^{t_i+T} \delta(s - x(t)) dt \qquad (2)$$

where δ is a Dirac delta function, $T \in \mathbb{R}^+$ is the time horizon, $t_i \in \mathbb{R}^+$ is the ith sampling time, and $s \in \mathbb{R}^v$ is the search domain. Define a target distribution $\phi(s) : \mathcal{X}_v \to \mathbb{R}^+$ with respect to which the robot is to be ergodic (i.e., time spent during a trajectory $x(t)$ is proportional to the spatial statistics of that region). An ergodic metric [20] which relates the two distributions $c(s, x(t))$ and $\phi(s)$ is:

$$\mathcal{E}(x(t)) = q \sum_{k \in \mathbb{N}^v} \Lambda_k \left(c_k - \phi_k\right)^2 \qquad (3)$$

$$= q \sum_{k \in \mathbb{N}^v} \left(\frac{1}{T} \int_{t_i}^{t_i+T} F_k(x(t)) dt - \phi_k \right)^2$$

where

$$\phi_k = \int_{\mathcal{X}_v} \phi(s) F_k(s) ds$$

and c_k are the Fourier decompositions[1] of $c(s, x(t))$ and $\phi(s)$ with

$$F_k(x) = \frac{1}{h_k} \prod_{i=1}^{v} \cos\left(\frac{k_i \pi x_i}{L_i} \right)$$

being the cosine basis function for a given coefficient $k \in \mathbb{N}^v$, h_k is a normalization factor defined in [20], and $\Lambda_k = (1 + \|k\|^2)^{-\frac{v+1}{2}}$ are weights on the frequency coefficients. A robot whose control inputs result in a trajectory $x(t)$ that minimizes (3) as $t \to \infty$ is then said to be optimally ergodic with respect to the target distribution.

Because we are computing the ergodic control in receding horizon, and the target distribution $\phi(s)$ can be time-varying, a history of where a robot has been is maintained in memory in order to compute the ergodic metric. The ergodic metric is then computed by adding a time parameter $\Delta t_\mathcal{E}$ which governs how far into the past the robot must remember where it has been. Equation (3) then becomes

$$\mathcal{E}(x(t)) = q \sum_{k \in \mathbb{N}^v} \left(\frac{1}{T} \int_{t_i - \Delta t_\mathcal{E}}^{t_i+T} F_k(x(t)) dt - \phi_k \right)^2. \qquad (4)$$

B. Ergodic Control

In [23] the ergodic controller is formulated using a trajectory optimizations scheme. While this approach does give optimal solutions, it is difficult for the controller to run in real time. As a result, [11] developed a hybrid systems approach using [22] to obtain control policies that sufficiently reduce the ergodic metric. We formulate our controller using a similar approach, but provide a variation to the controller that allows the policy to be fully distributable.

Rather than directly minimizing (3) with respect to $x(t)$ and $u(t)$, we consider the sensitivity of (3) with respect to an

[1]The cosine basis function is used, however, any choice of basis function F_k can be used.

infinitesimal application $\lambda \to 0$ of the best possible control $u_\star(t) : \mathbb{R}^+ \to \mathbb{R}^m$ that sufficiently reduces (3) at time τ from some default control $u_{\text{def}}(t) : \mathbb{R}^+ \to \mathbb{R}^m$. Following [11], we take the derivative of (3) with respect to the application duration λ which gives the sensitivity (known as the mode insertion gradient [24], [25], [26], [27]).

Proposition 1: The first order sensitivity of (3 with respect to the control duration λ of the applied control $u_\star(\tau)$ is

$$\left. \frac{\partial \mathcal{E}}{\partial \lambda} \right|_\tau = \rho(\tau)^T (f_2(\tau, \tau) - f_1(\tau)) \qquad (5)$$

where $f_2(t, \tau) = f(x(t), u_\star(\tau))$, $f_1(t) = f(x(t), u_{\text{def}}(t))$, and $\rho(t) : \mathbb{R}^+ \to \mathbb{R}^n$ is given by the differential equation

$$\dot{\rho} = -2\frac{q}{T} \sum_{k \in \mathbb{N}^v} \Lambda_k (c_k - \phi_k) \frac{\partial F_k}{\partial x} - \frac{\partial f}{\partial x}^T \rho(t)$$

with $\rho(t_i + T) = \mathbf{0} \in \mathbb{R}^n$.

Proof: See Appendix or [11]. ■

The mode insertion gradient now represents the sensitivity of the ergodic metric with respect to an application of a control $u_\star(t)$.

Given the mode insertion gradient, we seek to find the control $u_\star(t)$ that provides the most significant decrease in the objective (3). To do so, we define a secondary objective function

$$J_2 = \int_{t_i}^{t_i+T} \left. \frac{\partial \mathcal{E}}{\partial \lambda} \right|_t + \frac{1}{2} \|u_\star(t) - u_{\text{def}}(t)\|_R^2 \qquad (6)$$

where $R \in \mathbb{R}^{m \times m}$ is a positive definite matrix that weighs $u_\star(t)$.

Proposition 2: The solution to $u_\star(t)$ that minimizes (6) is

$$u_\star(t) = -R^{-1} h(x)^T \rho(t) + u_{\text{def}}(t). \qquad (7)$$

Proof: Taking the derivative of (6) with respect to control $u_\star(t)$ and setting the solution to zero gives

$$\frac{\partial J_2}{\partial u_\star} = \int_{t_i}^{t_i+T} \frac{\partial}{\partial u_\star} \left(\frac{\partial \mathcal{E}}{\partial \lambda} \right) + R(u_\star - u_{\text{def}}) dt$$

$$= \int_{t_i}^{t_i+T} h(x)^T \rho + R(u_\star - u_{\text{def}}) dt = 0 \qquad (8)$$

where the dependency on time is dropped for simplicity. Solving for $u_\star$ in (8) gives

$$u_\star(t) = -R^{-1} h(x(t))^T \rho(t) + u_{\text{def}}(t).$$

■

Lemma 1: Assuming that $h(x)^T \rho \neq 0$, the mode insertion gradient in (5) is always negative for $u_\star(t)$ defined in (7), that is $\frac{\partial \mathcal{E}}{\partial \lambda} < 0 \, \forall u_\star \in \mathcal{U}$ where $\mathcal{U}$ is the control space.

Proof: Inserting (7) into (5) gives

$$\frac{\partial \mathcal{E}}{\partial \lambda} = h(x)^T \rho \left(-R^{-1} h(x)^T \rho \right)$$

$$= -\rho^T h(x) R^{-1} h^T \rho = -\|h(x)^T \rho\|_{R^{-1}}^2 < 0. \qquad (9)$$

Thus (9) shows us that $\forall u_\star \in \mathcal{U}$ defined in (7), $\frac{\partial \mathcal{E}}{\partial \lambda} < 0$. ■ Because (7) always provides a negative $\frac{\partial \mathcal{E}}{\partial \lambda}$, this implies that each control that is chosen will result in a decrease in (3); thus eventually minimizing the ergodic metric. Additionally,

as in [11], a contractive constraint on the reduction of the ergodic metric is enforced that further provides a reduction in the ergodic metric from the previous control calculation time.

In many robotics applications, it is required that the control is saturated due to actuation limits in the robot while maintaining some form of sufficient decrease in the objective cost. In this work, we select a time of application τ that results in the most negative mode insertion gradient, or more formally written by

$$\tau_\star = \operatorname*{argmin}_\tau \frac{\partial \mathcal{E}}{\partial \lambda}.$$

A line search [28] is then used to find the duration λ that significantly reduces (3) subject to the saturated control $u_\star(\tau)$. The resulting control is then added to the default control $u_{\text{def}}(t) = u_\star(\tau) \forall t \in [\tau, \tau + \lambda] \cap [t_i, t_i + t_s]$ where t_s is the sampling time and $u_\star(\tau)$ is saturated.

The following subsection derives the ergodic control policy for decentralized multi-agent systems.

C. Decentralized Ergodic Control using Consensus

Consider a set of N connected agents with state $x(t) = \left[x_1(t)^\top, x_2(t)^\top, \ldots, x_N(t)^\top \right]^\top : \mathbb{R}^+ \to \mathbb{R}^{nN}$.

Proposition 3: Given the default trajectory of each agent $x(t) \forall t \in [t_i - \Delta t_\mathcal{E}, t_i + T]$ subject to $u_{\text{def}}(t)$, the control policy (7) is distributable amongst each individual agent and independent of the other agent's control policy.

Proof: Let us first define the dynamics of the collective multi-agent system as

$$
\begin{aligned}
\dot{x} = f(x, u) &= g(x) + h(x)u \\
&= \begin{bmatrix} g_1(x_1) \\ g_2(x_2) \\ \vdots \\ g_N(x_N) \end{bmatrix} + \begin{bmatrix} h_1(x_1) & \cdots & 0 \\ \vdots & \ddots & \\ 0 & & h_N(x_N) \end{bmatrix} u
\end{aligned} \quad (10)
$$

where $h(x)$ is block diagonal. The multi-agent system's contribution to the time-averaged statistics c_k can be rewritten as

$$
\begin{aligned}
c_k &= \frac{1}{N} \sum_{j=1}^N \frac{1}{T} \int_{t_i - \Delta\mathcal{E}}^{t_i + T} F_k(x_j(t)) dt \\
&= \frac{1}{T} \int_{t_i - \Delta\mathcal{E}}^{t_i + T} \tilde{F}_k(x(t)) dt
\end{aligned} \quad (11)
$$

where $\tilde{F}_k(x(t)) = \frac{1}{N} \sum_j F_k(x_j(t))$. The mode insertion gradient (18) under a multi-agent dynamical system now has $f_1(t)$ and $f_2(t, \tau)$ defined by (10) and the convolution equation for the adjoint variable $\rho(t)$ becomes

$$\dot{\rho} = -2\frac{q}{T} \sum_{k \in \mathbb{N}^v} \Lambda \left(c_k - \phi_k \right) \frac{\partial \tilde{F}_k}{\partial x} - \frac{\partial f}{\partial x}^\top \rho \quad (12)$$

where

$$\frac{\partial \tilde{F}_k}{\partial x} = \frac{1}{N} \begin{bmatrix} \frac{\partial F_k(x_1)}{\partial x_1} \\ \vdots \\ \frac{\partial F_k(x_N)}{\partial x_N} \end{bmatrix}$$

and

$$\frac{\partial f}{\partial x} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & 0 & \cdots & 0 \\ 0 & \frac{\partial f_2}{\partial x_2} & & \\ \vdots & & \ddots & \\ 0 & & & \frac{\partial f_N}{\partial x_N} \end{bmatrix}$$

is block diagonal. Because each agent's dynamics are independent of each other, (12) can be written independently for each agent as

$$\dot{\rho}_j = -2\frac{q}{TN} \sum_{k \in \mathbb{N}^v} \Lambda_k(c_k - \phi_k) \frac{\partial F_k(x_j)}{\partial x_j} - \frac{\partial f_j}{\partial x_j}^\top \rho_j.$$

Similarly, the ergodic control policy derived from (8) becomes

$$
\begin{aligned}
\begin{bmatrix} u_{\star,1}(t) \\ \vdots \\ u_{\star,N}(t) \end{bmatrix} &= -R^{-1} \begin{bmatrix} h_1(x_1) & & \\ & \ddots & \\ & & h_N(x_N) \end{bmatrix}^T \begin{bmatrix} \rho_i(t) \\ \vdots \\ \rho_N(t) \end{bmatrix} \\
&+ \begin{bmatrix} u_{\text{def},1}(t) \\ \vdots \\ u_{\text{def},N}(t) \end{bmatrix}
\end{aligned} \quad (13)
$$

where $R \in \mathbb{R}^{mN \times mN}$ and mN is the size of the collective multi-agent system control input. Since $h(x)$ is block diagonal, (13) becomes

$$u_{\star,j}(t) = -R_j^{-1} h_j(x_j)^T \rho_j(t) + u_{\text{def},j}(t) \quad (14)$$

for each agent $j \in [1, \ldots, N]$ and $R_j \in \mathbb{R}^{m \times m}$. The control policy in (14) for the jth agent does not depend on the ith agent and therefore is distributable. ∎

While the jth control policy is independent of the ith control policy, it is assumed starting from (4) that each agent's past and anticipated trajectory is known to all agents before calculating the control policy. We can consider this a distributed ergodic control policy where the control computation is still done on individual CPUs on-board the agents, but the initial conditions are required to be sent from a central communication hub. Instead of a distributed controller, we seek to completely remove the need for a centralized communication hub and have fully independent agents. We solve this problem using consensus-based methods where a network of agents communicates with one another.

Rather than communicating the past and anticipated trajectories of each agent (which may have large dimensionality) in the network, we communicate the c_k values instead. [2]

Proposition 4: A connected multi-agent network under consensus over the c_k coefficients approximates the time-average statistics c_k of the centralized ergodic metric (11), that is $\tilde{c}_k \to c_k$ as $t \to \infty$ where $\tilde{c}_k$ is the consensus-based time-average statistics.

[2]It is assumed that each agent as the same ϕ_k target, however, the same analysis can be done to form a consensus on the target ϕ_k values.

Proof: Consider the collective time-averaged statistics c_k for the system in (11):

$$c_k = \frac{1}{N} \sum_{j=1}^{N} \frac{1}{T} \int_{t_i}^{t_i+T} F_k(x_j(t)) dt.$$

Note that equation (11) is simply an averaging of the individual agent's spatial statistics. Let us then define a row and column stochastic consensus matrix P (e.g., $\sum_j P_{ij} = \mathbf{1}$) that defines the network connectivity amongst the agents [29], [30]. The operation $\sum_j P_{ij}c_{k,j}$ is equivalent to taking an average of the local $c_{k,j}$ values for each neighboring agent. [3] Therefore, we can write a consensus on the collective c_k (11) using P as [29], [30]

$$\lim_{t_k \to \infty} \sum_j P_{ij}^{t_k} c_{k,j} = \frac{1}{N} \sum_j \frac{1}{T} \int_{t_0}^{t_0+T} F_k(x_j(t)) dt.$$

where N is the number of agents, t_k is the number of times that $P_{ij}c_{k,j}$ values have been communicated through the network and averaged. Thus consensus amongst all the agents approximates the collective multi-agent system time-averaged statistics c_k in (11). ∎

Choosing the time of application and duration remains the same where each agent decides their own actions according to the decentralized mode-insertion gradient. Algorithm 1 is provided to illustrate the decentralized ergodic control for multi-agent systems.

Algorithm 1 Decentralized Ergodic Control
___
1: **initialize:** agents N with initial condition $x_j(0)$, initial target distribution $\phi_{k,0}$, t_0, t_f, t_s, time horizon T, ergodic memory $\Delta t_{\mathcal{E}}$ and network P.
2: **while** $t_i < t_f$ **do**
3:     **for** agent in agents **do**   ▷ Control step
4:        simulate $x(t), \rho(t)$ for $t \in [t_i, t_i + T]$ from $x(t_i)$
5:        compute $u_\star(\tau)$ from (7)
6:        calculate τ and λ from [22], [11]
7:        $u_{\text{def}}(t) = u_\star(\tau) \forall t \in [\tau, \tau + \lambda] \cap [t_i, t_i + t_s]$
8:     **end for**
9:     **for** agent in agents **do**   ▷ Communication Loop
10:        Send $c_{k,j}$ to i^{th} neighbors in the network P
11:        Receive $c_{k,i}$ from neighbors and average amongst i^{th} neighbors
12:     **end for**
13:     apply control $u_{\text{def}}(t_i)$
14:     $i \leftarrow i + 1$
15: **end while**
___

III. Ergodic Area Coverage for Multi-Agent Elevation Mapping

In this section we illustrate the capabilities of a decentralized ergodic controller for multi-agent area coverage for

---

[3]For simplicity in notation, we assume that P_{ij} refers to a block matrix such that $P \in \mathbb{R}^{|k|N \times |k|N}$ and $P_{ij} \in \mathbb{R}^{|k| \times |k|}$ where $|k|$ is the total number of c_k coefficients.

---

elevation mapping. We use this example to show improved area coverage of a decentralized ergodic controller while comparing with a centralized controller for the same task.

A. Problem Setup

A 12 dimensional quadrotor [31] is used for the robotic agent dynamics with 4 inputs directly controlling thrust, yaw, pitch, and roll angular accelerations. Each agent measures ground elevation relative to the agent's altitude which it uses to construct a model of the terrain. Three agents are used that are fully connected to one another. The agents are randomly initialized and a Gaussian Process [32], [33] is used to construct the terrain elevation from data collected after 10 second intervals.

B. Results

Figure 1 illustrates the algorithm for area coverage of three decentralized robotic agents, For comparison purposes, the area coverage of a single agent under the ergodic control policy is shown. Due to the sharing of where each agent intends to go and where they have been which results in a much more efficient search as each agent chooses the best possible action that helps reduce the ergodic metric. Compared to a single agent, the terrain estimate metric for multi-agent area coverage is significantly less. The ergodic control automatically takes into account dynamic constraints and the histories of the other agents in order to allocate where each agent should go in a decentralized fashion. We see this in Fig. 1(c) where the multi-agent system immediately acquires a sufficient terrain model within the first ten seconds according the error norm on the estimate.

A comparison is in Fig. 2 with respect to a centralized hub multi-agent ergodic system. Not much performance is lost lost within the first 5 seconds of the algorithm when the robotic agents are still trying to achieve a consensus. After consensus is close to convergence, the decentralized ergodic policy functions just as well as the centralized ergodic policy.

IV. Decentralized Ergodic Control for Multi-Agent Target Localization

In this section, decentralized ergodic control is used for multi-agent target localization. We use the example of multi-agent target localization because this platform provides us with novel demonstration of the decentralized ergodic control algorithm through a well known robotics problem. Moreover, we use this application to demonstrate the generalizability of the algorithm.

A. Problem Setup

The goal of target localization is to have the agents locate the target (or targets) in the environment. Bearing only sensors [11], [34] are used for sensing the target with the same three agents as mentioned in Section III with quadcopter dynamics. The obstacles are incorporated into the objective with an obstacle avoidance cost which we define by the function $\Theta(x) : \mathbb{R}^n \to \mathbb{R}^+$ which is a direct penalty if the agent goes near an obstacle (see Appendix for more
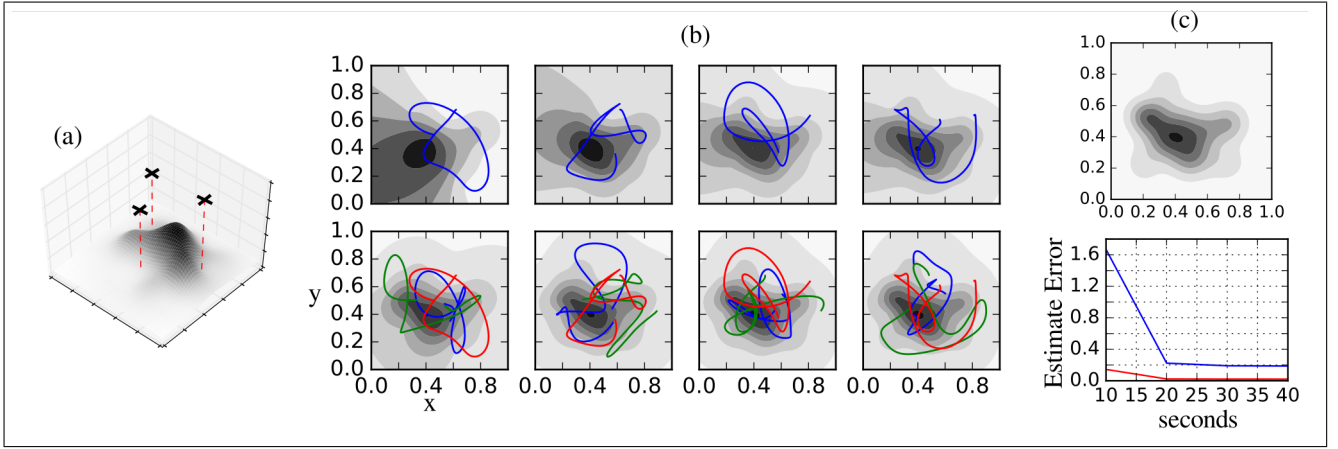
Fig. 1. (a) Three quadcopter agents are depicted in the map with the terrain. The red dashed line indicates the location. (b) Trajectories of a single agent (blue) and a multi-agent system (blue, red, green) are shown estimating terrain. The terrain map is obtained from height measurements (dark regions represent high elevation). (c) Top-down orthographic view of the terrain map for comparison with the results in (b). Error of the estimate is shown at 10 second intervals of collected data. The more area coverage, the more collected data and a resulting better terrain map.
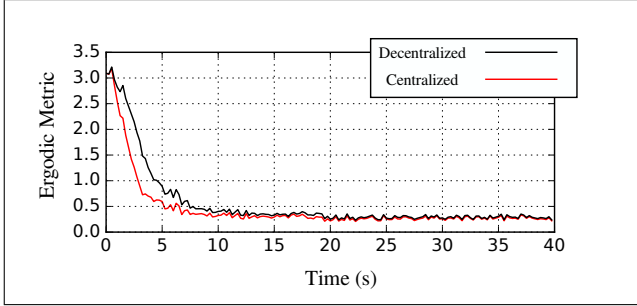

Fig. 2. Comparison of ergodic metric for a decentralized ergodic scheme versus a centralized ergodic scheme. Due to communication and consensus amongst the agent in a decentralized scheme, the ergodic metric does not reduce as quickly as a centralized scheme would. However, the decentralized scheme is quick to reach consensus and performs comparably to the centralized scheme.

detail on augmenting the controller with the obstacle cost). In addition, we constrain the radius of the target sensor to 0.38 meter diameter, thus limiting the total area coverage from the sensor. The targets are uniformly dispersed throughout the terrain of size $[0, 1] \times [0, 1]$ [4] such that they do not intersect with the obstacles. Targets are localized using an extended Kalman filter (EKF) [35], [36]. The ergodic controller is initialized with a uniform target distribution. The prior on the targets is initialized as uniform over the search terrain and a distributed EKF updates the prior for the network system [37]. The target distribution is given by the expected information density [6], [11]

$$\phi(s) = \eta \det \left[\int_{\theta} \frac{\partial \left(\Upsilon(\theta, s) \right)^{T}}{\partial \theta} \Sigma^{-1} \frac{\partial \left(\Upsilon(\theta, s) \right)}{\partial \theta} p(\theta) d\theta \right] \tag{15}$$

where η is a normalization factor, $\Upsilon(\theta, s)$ is the bearing only measurement model parametrized by the position of the targets θ.

[4]This is can be easily adjusted in experimentation if the terrain is much larger.

B. Results

Figure 3 shows the trajectories of the 3 agents localizing 4 targets in the environment. Each each agent chooses a different path that reduces the ergodic measure as well as increases the area coverage. The agents each localize the targets while successfully avoiding obstacles (illustrated as the black colored squares). In Fig. 3(c), the agents collectively localize each target within the first 10 seconds in a fully decentralized manner. While the outcome of the results are unsurprising, it is interesting to see the automatic collaborative nature of a decentralized ergodic controller in the presence of constraints (i.e., dynamics, obstacles, and sensors). Simple augmentations to the algorithm can include multiple objectives [38], [39] and consensus on the target distribution ϕ_k values. Moreover, improving network quality is also possible for noisy multi-agent networks with connectivity drop-out [40].

In the following subsection we take a second glance at the ergodic policy from a game theoretic perspective.

V. ERGODIC CONTROL POLICIES AS NASH EQUILIBRIUM STRATEGIES

In this section, we analyze the ergodic control policy from a game theoretic point of view in adversarial multi-agent games.

Definition 1: A game is defined by a tuple $(\mathcal{P}, \mathcal{A}, \mathcal{O}, \mu, \mathcal{U})$ where $\mathcal{P}$ is the set of players in a game, $\mathcal{A}$ is the set of control actions $u(t)$ where $u_i(t) \forall t \in [t_i, t_i + T], \forall i \in \mathcal{P}$ is considered an action or strategy profile, $\mathcal{O}$ is the set of outcomes (or state trajectories in our case), $\mu : \mathcal{A} \to \mathcal{O}$ is the function that maps actions to outcomes (in our case this is the robot dynamics), and last $\mathcal{U} : \mathcal{O} \to \mathbb{R}$ is a utility function for each player.

Each agent defined by $\mathcal{P}$. The action profile or strategy $\mathcal{A}$ is defined by the ergodic control policy subject to a target distribution. The resultant trajectory $x(t)$ for each agent is the outcome $\mathcal{O}$ subject to the actions passing through the dynamics $f(x, u)$ of the system (μ). Here, we treat the utility
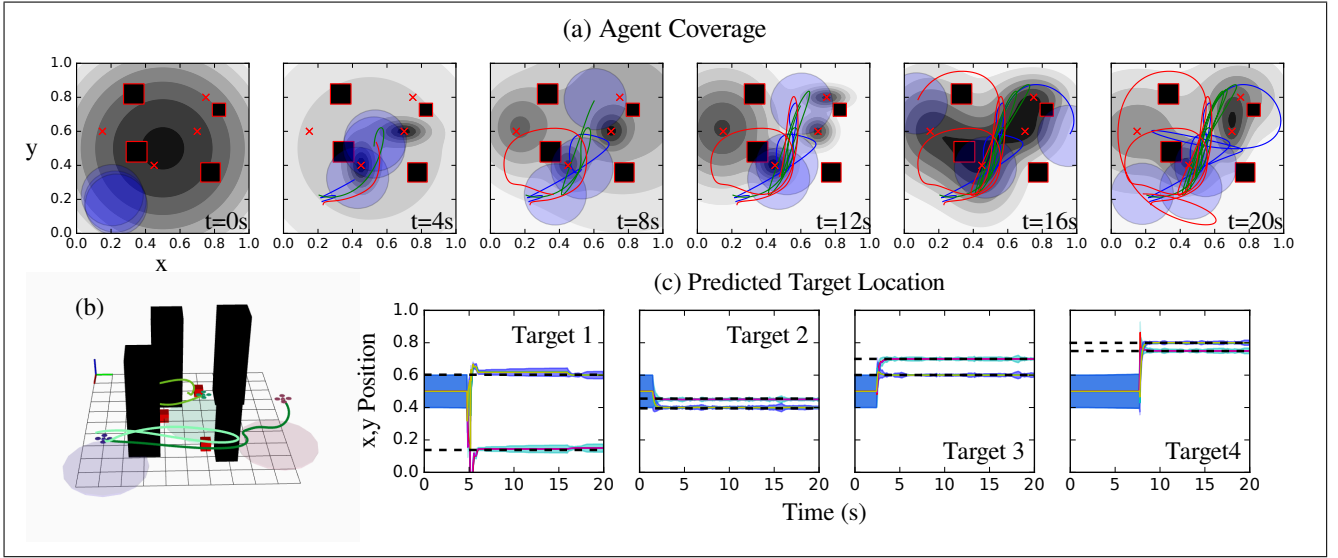
Fig. 3. Target localization is illustrated using 3 agents with quadcopter dynamics and 4 unknown stationary targets depicted as the red crosses. (a) Area coverage of the quadcopter is shown as the lines associated with each quadcopter at distinct times. The density function underneath shows the likely locations of the targets with darker regions indicating higher likelihood values. Obstacles are shown as black squares with red outlines. A 3D rendering of the environment is shown in (b). Each agent only has a field of range of 0.36 meters as shown as the transparent blue circles. Within 10 seconds the agents under decentralized ergodic policies are able to provide a consensus on the location of the targets. (c) Extended Kalman filter values on the location of the target is shown. The black dashed line is the target location ground truth. We refer the reader to the attached multimedia of the target localization example.

function $\mathcal{U}$ as the ergodic metric. In game theory the notion of Nash equilibrium [41], [42] is often used to describe a strategy in a game.

Definition 2: A strategy is a Nash equilibrium if for each agent i, $\mathcal{U}_i(u) \leq \mathcal{U}_i(u_{-i})$ where u_{-i} is the updated strategy profile for all agents not including agent i's strategy.

Nash equilibrium tells us whether a strategy results in the best possible expected utility of each agent subject to the other agents' actions. We consider Nash equilibrium in the problem of target localization and evasion. Specifically, we look at what strategy an evader can use to acquire a Nash equilibrium with the pursuer (localizer) (i.e., a game between the pursuer and evader while the pursuer expends energy not localizing the evader).

Theorem 1: A Nash equilibrium strategy against a pursuer with an ergodic policy is for the evader to adopt an ergodic policy.

Proof: Consider two agents, a and b on opposing sides of a game. Agent a is ergodic with respect to a target distribution $\phi_a(s)$. Agent b is ergodic with respect to $\phi_b(s)$. We assume that the target distribution of agent a and b is a function of the state of the agents, that is, $\phi_a(s) = \phi_a(s, x_a(t), x_b(t))$ and $\phi_b(s) = \phi(s, x_a(t), x_b(t))$. From Lemma 1, we have shown that $\frac{\partial \mathcal{E}}{\partial \lambda} < 0$, $\forall u(t)$ defined by an ergodic policy. As a result, we can assume that as $t \to \infty$, both agents are asymptotically optimally ergodic with respect to their own target distributions so long as each action reduces the ergodic objective. Therefore, we can write the change in the utility function—which we define as the

ergodic metric—as

$$\mathcal{U}_i(x(t) \mid u) - \mathcal{U}_i(x(t) \mid u_{-i}) = \Delta U_i \approx \frac{\partial \mathcal{E}_i}{\partial \lambda}\lambda < 0$$
$$\mathcal{U}_i(x(t) \mid u) - \mathcal{U}_i(x(t) \mid u_{-i}) < 0$$
$$\mathcal{U}_i(x(t) \mid u) < \mathcal{U}_i(x(t) \mid u_{-i})$$

Thus, an ergodic control strategy is a Nash equilibrium strategy. ∎

This kind of analysis lends some insight towards formally viewing ergodic policies with respect to game theory and with application in general multi-agent games.

VI. CONCLUSIONS

We present a fully decentralized formulation of ergodic control for multi-agent systems. It is shown that this formulation is distributable and under consensus, decentralized. Examples of area coverage and target localization and evasion illustrate the application of the algorithm for multi-agent systems. In these examples, the algorithm is shown to synthesize control actions that simultaneously consider the multi-agent network as well as the individual agent dynamic constraints. Last, we provide a game theoretic analysis of the algorithm such that we prove it is the best possible control policy that an agent can adopt in a multi-agent game. This opens up the possibilities of adapting other multi-agent objectives—such as target localization—into a decentralized network.

In future work will examine the effects of the consensus matrix on the decentralized ergodic policy and augmentations to the target distribution to maintain network connectivity for time-varying network topologies.

APPENDIX

A. Proof

Here, the proof of Proposition 1 is as follows.

Proof: Taking the derivative of (3) with respect to λ gives us

$$\frac{\partial \mathcal{E}}{\partial \lambda} = 2\frac{q}{T}\sum_{k\in\mathbb{N}^v}\Lambda_k(c_k-\phi_k)\frac{\partial}{\partial\lambda}\int_\tau^{t_0+T}F_k(x(t))dt \quad (16)$$

$$= 2\frac{q}{T}\sum_{k\in\mathbb{N}^v}\Lambda_k(c_k-\phi_k)\int_\tau^{t_0+T}\frac{\partial F_k(x(t))}{\partial x}\frac{\partial x(t)}{\partial\lambda}dt.$$

From [26], [27] and that

$$\frac{\partial x(t)}{\partial\lambda} = \Phi(t,\tau)\left(f_2(\tau,\tau)-f_1(\tau)\right), \quad (17)$$

where $\Phi(t,\tau)$ is the state transition matrix, (16) is written as

$$\left.\frac{\partial\mathcal{E}}{\partial\lambda}\right|_\tau = \rho(\tau)^T\left(f_2(\tau,\tau)-f_1(\tau)\right), \quad (18)$$

where

$$\rho(\tau) = 2\frac{q}{T}\sum_{k\in\mathbb{N}^v}\Lambda_k(c_k-\phi_k)\int_\tau^{t_0+T}\frac{\partial F_k(x(t))}{\partial x}\Phi(t,\tau)dt. \quad (19)$$

We can see that (19) is a convolution equation for

$$\dot{\rho} = -2\frac{q}{T}\sum_{k\in\mathbb{N}^v}\Lambda_k(c_k-\phi_k)\frac{\partial F_k(x)}{\partial x}-\frac{\partial f}{\partial x}^T\rho \quad (20)$$

where $\rho\in\mathbb{R}^n$, $\rho(t_0+T)=\mathbf{0}\in\mathbb{R}^n$. Thus, the mode insertion gradient is given as

$$\left.\frac{\partial\mathcal{E}}{\partial\lambda}\right|_\tau = \rho(\tau)^T(f_2(\tau,\tau)-f_1(\tau)) \quad (21)$$

subject to

$$\dot{\rho} = -2\frac{q}{T}\sum_{k\in\mathbb{N}^v}\Lambda_k(c_k-\phi_k)\frac{\partial F_k(x)}{\partial x}-\frac{\partial f}{\partial x}^T\rho. \quad (22)$$

$\blacksquare$

B. Augmented Objective for Obstacle Avoidance

Here we present an augmented objective function for ergodic exploration and obstacle avoidance. Assuming we know that a penalty function $\Theta(x):\mathbb{R}^n\to\mathbb{R}^+$ we can define an objective cost

$$J(x(t) = q_1\mathcal{E}(x(t)) + q_2\int_{t_i}^{t_i+T}\Theta(x(t))dt \quad (23)$$

where q_1, q_2 are weights on the ergodic cost and the obstacle avoidance cost. Following the formulation of the ergodic control policy, the adjoint variable $\rho(t)$ differential equation for the mode insertion gradient (5) becomes

$$\dot{\rho} = -2\frac{q_1}{T}\sum_{k\in\mathbb{N}^v}\Lambda_k(c_k-\phi_k)\frac{\partial F_k}{\partial x}-q_2\frac{\partial\Theta}{\partial x}-\frac{\partial f}{\partial x}^\top\rho. \quad (24)$$

The control formulation remains the same as done in Section II.

REFERENCES

[1] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping SLAM: Part II," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 108–117, 2006.

[2] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE robotics & automation magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[3] H. Choset and K. Nagatani, "Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 2, pp. 125–137, 2001.

[4] I. Abraham, A. Prabhakar, M. J. Hartmann, and T. D. Murphey, "Ergodic exploration using binary sensing for nonparametric shape estimation," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 827–834, 2017.

[5] G. De La Torre, K. Flaßkamp, A. Prabhakar, and T. D. Murphey, "Ergodic exploration with stochastic sensor dynamics," in *American Control Conference (ACC), 2016*, 2016, pp. 2971–2976.

[6] L. M. Miller, Y. Silverman, M. A. MacIver, and T. D. Murphey, "Ergodic exploration of distributed information," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 36–52, 2016.

[7] Y. Yang, H. Zhang, Z.-H. Lin, Y. S. Zhou, Q. Jing, Y. Su, J. Yang, J. Chen, C. Hu, and Z. L. Wang, "Human skin based triboelectric nanogenerators for harvesting biomechanical energy and as self-powered active tactile sensor system," *ACS nano*, vol. 7, no. 10, pp. 9213–9222, 2013.

[8] R. S. Dahiya, G. Metta, M. Valle, and G. Sandini, "Tactile sensing: From humans to humanoids," *IEEE Transactions on Robotics*, vol. 26, no. 1, pp. 1–20, Feb 2010.

[9] F. Iida and A. J. Ijspeert, "Biologically inspired robotics," in *Springer Handbook of Robotics*. Springer, 2016, pp. 2015–2034.

[10] L. Zhu, L. Zeng, X. Chen, X. Luo, and X. Li, "A bioinspired touching sensor for amphibious mobile robots," *Advanced Robotics*, vol. 29, no. 22, pp. 1437–1452, 2015.

[11] A. Mavrommati, E. Tzorakoleftherakis, I. Abraham, and T. D. Murphey, "Real-time area coverage and target localization using receding-horizon ergodic exploration," *IEEE Transactions on Robotics*, vol. 34, no. 1, pp. 62–80, 2018.

[12] L. Mihaylova, T. Lefebvre, H. Bruyninckx, K. Gadeyne, and J. De Schutter, "A comparison of decision making criteria and optimization methods for active robotic sensing," in *International Conference on Numerical Methods and Applications*. Springer, 2002, pp. 316–324.

[13] A. Ryan and J. K. Hedrick, "Particle filter based information-theoretic active sensing," *Robotics and Autonomous Systems*, vol. 58, no. 5, pp. 574–584, 2010.

[14] D. Carmel and S. Markovitch, "Exploration strategies for model-based learning in multi-agent systems: Exploration strategies," *Autonomous Agents and Multi-agent systems*, vol. 2, no. 2, pp. 141–172, 1999.

[15] G. Dudek, M. R. Jenkin, E. Milios, and D. Wilkes, "A taxonomy for multi-agent robotics," *Autonomous Robots*, vol. 3, no. 4, pp. 375–397, 1996.

[16] C. Manss, D. Shutin, T. Wiedemann, A. Viseras, and J. Mueller, "Decentralized multi-agent entropy-driven exploration under sparsity constraints," in *Compressed Sensing Theory and its Applications to Radar, Sonar and Remote Sensing (CoSeRa)*, 2016, pp. 143–147.

[17] A. Viseras Ruiz, M. Angermann, I. Wieser, M. Frassl, and J. Mueller, "Efficient multi-agent exploration with Gaussian processes," 2014.

[18] M. A. Khamis and W. Gomaa, "Adaptive multi-objective reinforcement learning with hybrid exploration for traffic signal control based on cooperative multi-agent framework," *Engineering Applications of Artificial Intelligence*, vol. 29, pp. 134–151, 2014.

[19] F. Pei, M. Wu, and S. Zhang, "Distributed SLAM using improved particle filter for mobile robot localization," *The Scientific World Journal*, 2014.

[20] G. Mathew and I. Mezić, "Metrics for ergodicity and design of ergodic dynamics for multi-agent systems," *Physica D: Nonlinear Phenomena*, vol. 240, no. 4, pp. 432–442, 2011.

[21] D. A. Shell and M. J. Matarić, "Ergodic dynamics for large-scale distributed robot systems," in *International Conference on Unconventional Computation*. Springer, 2006, pp. 254–266.

[22] A. R. Ansari and T. D. Murphey, "Sequential action control: Closed-form optimal control for nonlinear and nonsmooth systems," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1196–1214, 2016.

[23] L. M. Miller and T. D. Murphey, "Trajectory optimization for continuous ergodic exploration," in *American Control Conference*, 2013, pp. 4196–4201.

[24] R. Vasudevan, H. Gonzalez, R. Bajcsy, and S. S. Sastry, "Consistent approximations for the optimal control of constrained switched systems—part 1: A conceptual algorithm," *SIAM Journal on Control and Optimization*, vol. 51, no. 6, pp. 4463–4483, 2013.

[25] H. Axelsson, Y. Wardi, M. Egerstedt, and E. Verriest, "Gradient descent approach to optimal mode scheduling in hybrid dynamical systems," *Journal of Optimization Theory and Applications*, vol. 136, no. 2, pp. 167–186, 2008.

[26] M. Egerstedt, Y. Wardi, and H. Axelsson, "Transition-time optimization for switched-mode dynamical systems," *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 110–115, 2006.

[27] T. Caldwell and T. Murphey, "Projection-based iterative mode scheduling for switched systems," *Nonlinear Analysis: Hybrid Systems*, vol. 21, pp. 59–83, 2016.

[28] L. Armijo, "Minimization of functions having lipschitz continuous first partial derivatives," *Pacific Journal of mathematics*, vol. 16, no. 1, pp. 1–3, 1966.

[29] N. Deo, *Graph theory with applications to engineering and computer science*. Courier Dover Publications, 2016.

[30] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: numerical methods*. Prentice hall Englewood Cliffs, NJ, 1989, vol. 23.

[31] P. Martin and E. Salaün, "The true role of accelerometer feedback in quadrotor control," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2010, pp. 1623–1629.

[32] S. Vasudevan, F. Ramos, E. Nettleton, and H. Durrant-Whyte, "Gaussian process modeling of large-scale terrain," *Journal of Field Robotics*, vol. 26, no. 10, pp. 812–840, 2009.

[33] C. Plagemann, S. Mischke, S. Prentice, K. Kersting, N. Roy, and W. Burgard, "Learning predictive terrain models for legged robot locomotion," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008, pp. 3545–3552.

[34] M. Deans and M. Hebert, "Experimental comparison of techniques for localization and mapping using a bearing-only sensor," in *Experimental Robotics VII*, 2001, pp. 395–404.

[35] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[36] S. J. Julier and J. K. Uhlmann, "New extension of the kalman filter to nonlinear systems," in *AeroSense'97*. International Society for Optics and Photonics, 1997, pp. 182–193.

[37] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri, "Distributed kalman filtering based on consensus strategies," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, 2008.

[38] J.-H. Cho, Y. Wang, R. Chen, K. S. Chan, and A. Swami, "A survey on modeling and optimizing multi-objective systems," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1867–1901, 2017.

[39] R. L. Moritz and S. Mostaghim, "Heterogeneous evolutionary swarms with partial redundancy solving multi-objective tasks," in *International Conference on Evolutionary Multi-Criterion Optimization*, 2017, pp. 453–468.

[40] A. C. Satici and M. W. Spong, "Global swarming while preserving connectivity via lagrange–poincarè equations," *Automatica*, vol. 71, pp. 369–380, 2016.

[41] S. Bhattacharya and S. Hutchinson, "On the existence of Nash equilibrium for a two player pursuit-evasion game with visibility constraints," in *Algorithmic Foundation of Robotics VIII*, 2009, pp. 251–265.

[42] R. B. Myerson, "Refinements of the Nash equilibrium concept," *International Journal of Game Theory*, vol. 7, no. 2, pp. 73–80, 1978.