

# Distributed Robot Ensemble Control for Deployment to Multiple Sites

Paper-ID 177

**Abstract**—We address the ensemble synthesis of distributed control policies to allocate a team of homogenous robots to a collection of spatially distributed tasks. We assume individual robot controllers are derived via the sequential composition of individual task controllers and develop an appropriate macroscopic description of the team dynamics. A feedback control strategy is synthesized using the macroscopic model to enable the team to maintain a desired distribution of robots across the various tasks while minimizing the variance of the robot population at each task. We present a distributed implementation of the proposed ensemble feedback strategy that can be implemented with minimal communication requirements. We establish stability properties of our ensemble controller and verify the feasibility of the distributed ensemble controller through high-fidelity simulations.

## I. INTRODUCTION

We address the dynamic allocation of a team of robots to a collection of spatially distributed tasks which is relevant for large scale environmental monitoring, surveillance, and automated warehouse distribution systems. In these applications, the team must have the ability to autonomously distribute among the various locales/tasks and redistribute to ensure task completion and/or coverage that may be affected by robot failures or changes in the environment. This is similar to the multi-task (MT), single-robots (SR), time-extended assignment (TA) problem [5]. In the multi-robot domain, market-based approaches [4, 3] have been successful and can be further improved when learning is incorporated [2]. Nevertheless, these methods often scale poorly in terms of team size and number of tasks [3, 8]. Furthermore, in applications where inter-agent wireless communication is limited, it is often difficult to devise reliable strategies to ensure timely communication of the various local costs and utilities required by these existing allocation approaches.

Different from existing work, we employ a macroscopic description of the ensemble dynamics to synthesize distributed agent-level control policies to dynamically allocate the team to the various tasks. In addition, our strategies will enable the team to shape both the mean and variance of the robot ensemble distribution. In recent years, macroscopic continuous models have been employed to describe the team dynamics of robotic self-assembly [11, 16] and robotic swarm systems [15, 14, 12]. These continuous population models are usually obtained by representing the individual robot controllers as probabilistic finite state machines and approximating the dynamics of a collection of discrete Markov processes as a continuous-time Markov process. The macroscopic models are then used to analyze the effects of microscopic, or agent-level,

behaviors on ensemble/team performance.

In this work, we consider the multi-site allocation problem, first presented in [9]. We assume a homogeneous ensemble of robots such that the agent-level control policies are obtained via the sequential composition of individual task controllers. Similar to [12, 1], we consider the design of stochastic transition rules to enable the team to autonomously achieve a desired distribution across the various tasks. Different from [12, 1], we synthesize agent-level control policies that can affect both the mean and the variance of the distribution across the various tasks. Our strategy is inspired by Klavins’s work on controlling the population dynamics of genes where a polynomial Stochastic Hybrid System (pSHS) is employed to describe the ensemble dynamics of the underlying agent-based system [13]. Here, we model the simultaneous execution of spatially distributed tasks by an ensemble of robots as a pSHS and employ moment closure techniques to model the moment dynamics of ensemble distribution [10]. The main contribution is a team-size invariant approach towards the design of distributed agent-level control policies that can affect both the mean and the variance of the robot ensemble distribution. As such, the approach has the ability to respond to robot failures in a natural way, ensuring graceful degradation.

The paper is structured as follows: We formulate our approach in Section II and describe the synthesis and analysis of our ensemble feedback strategy in Section III. Section IV describes the distributed implementation of our macroscopically derived feedback controller in the presence of full communication and when robots rely solely on local communication. Section V presents our simulation results. We conclude with a discussion of our results and directions for future work in Sections VI and VII respectively.

## II. PROBLEM FORMULATION

Consider the assignment of  $N$  robots to execute  $M$  tasks each located at a different locale within the workspace. The objective is to synthesize a decentralized control strategy that will enable the team to autonomously distribute across the  $M$  tasks and maintain the desired allocation at the various locales. Different from [12, 1], our goal is to design distributed agent-level control policies that can affect *both* the mean and the variance of the desired ensemble distribution.

### A. Individual Robot Controller

Given a collection of  $\{1, \dots, M\}$  tasks/sites, we use a directed graph,  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , to model the pairwise precedence constraints between the tasks. Each task is represented by a

vertex in  $\mathcal{V} = \{1, \dots, M\}$ . A directed edge exists between two vertices  $(i, j) \in \mathcal{V} \times \mathcal{V}$  if task  $i$  must precede task  $j$  and we denote this relation as  $i \prec j$ . Then, the set of edges,  $\mathcal{E}$ , is given by  $\mathcal{E} = \{\forall (i, j) \in \mathcal{V} \times \mathcal{V} | i \prec j\}$ . We assume  $\mathcal{G}$  is a strongly connected graph, *i.e.*, a directed path exists for any  $i, j \in \mathcal{V}$ .

Given the  $M$  tasks, we denote the set of task controllers for each task as  $\{U_1, \dots, U_M\}$  and assume that the single robot controller is obtained through the sequential composition of  $\{U_1, \dots, U_M\}$  such that the precedence constraints specified by  $\mathcal{G}$  are satisfied. We represent the robot controller as a finite state automaton where each automaton state  $i$  is associated with a task controller  $U_i$ . Fig. 1(a) shows an example robot controller where the arrows denote state transitions that satisfy the constraints specified in  $\mathcal{G}$ .

In this work, we consider the surveillance of  $M$  sites where the team must maintain some desired allocation of the robots across the various locations. Robots are tasked to monitor each site for a pre-specified amount of time, *i.e.*, execute  $U_i$ . Once the task is completed, robots must navigate to the next adjacent site based on the constraints encoded in  $\mathcal{G}$ . As such, we assume each robot has complete knowledge of  $\mathcal{G}$ , the ability to localize within the workspace, and is capable of navigating from one task/site to another while avoiding collisions with other robots in the workspace.

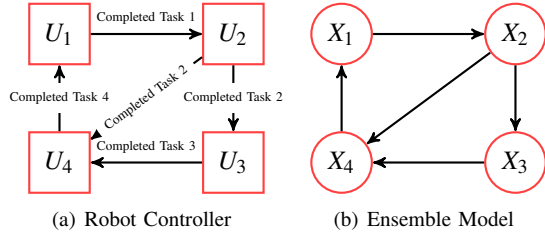


Fig. 1: (a) The robot controller. The robot changes controller states based on the guard conditions. (b) Graphical representation of the equivalent chemical reaction network for the team  $N$  robots distributed across the  $M$  tasks.

### B. The Ensemble Model

For a team of  $N$  robots, each executing the same sequentially composed controller, *e.g.*, the one in Fig. 1(a), the ensemble dynamics can be represented by an equivalent chemical reaction network. This abstraction allows us to model the multi-task/site allocation as a *polynomial stochastic hybrid system* (pSHS) and to use moment closure techniques to model the time evolution of the distribution of the team across the various tasks/sites.

Let  $X_i(t)$  and  $\bar{X}_i$  denote the number of robots executing task  $i$  or at site  $i$  and the desired number of robots for task  $i$  respectively. The system state is given by  $\mathbf{X}(t) = [X_1(t), \dots, X_M(t)]^T$  with the desired distribution of the ensemble given by  $\bar{\mathbf{X}} = [\bar{X}_1, \dots, \bar{X}_M]^T$ . Since the tasks are spatially distributed and robots must navigate from one site to another while avoiding collisions with other robots, we model the variability in robot arrival times at each task using *transition rates*. For every edge

$e_{ij} \in \mathcal{E}$ , we assign constant  $k_{ij} > 0$  such that  $k_{ij}$  defines the transition probability per unit time for one agent from site  $i$  to go to site  $j$ .

Given  $\mathcal{G}$  and the set of  $k_{ij}$ 's, we model the ensemble dynamics as a set of chemical reactions of the form:



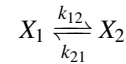
The above reaction represents a stochastic transition rule with  $k_{ij}$  as the per unit reaction rate and  $X_i(t)$  and  $X_j(t)$  as discrete random variables. In the robotics setting, equation (1) implies that robots at site  $i$  will transition to site  $j$  with a rate of  $k_{ij}X_i$ . Further, we assume the ensemble dynamics is Markov which will allow us to model the moment dynamics of the distribution as a set of linear differential equations. It is important to note that in general  $k_{ij} \neq k_{ji}$  and  $k_{ij}$  encodes the inverse of the average time a robot spends at task/site  $i$ .

1) *Moment Closure*: Given the set of reactions in (1), the moment equations for the discrete random variable  $X_i$  is given by the extended generator of the system [10]. For a real-valued function  $\psi(X_i)$ , the extended generator is an expression for the time derivative of the expected value of  $\psi$ , *i.e.*,  $\frac{d}{dt}E[\psi(X_i)] = E[L\psi(X_i)]$ , and takes the form

$$L\psi(X_i) = \sum_j [(\psi(X_i - 1) - \psi(X_i))w_{ji} + (\psi(X_i + 1) - \psi(X_i))w_{ij}]. \quad (2)$$

The right hand side of (2) gives the continuous time derivatives of the system for a discrete change in the state  $X_i$ . The expression  $[\psi(X_i - 1) - \psi(X_i)]$  represents the change in  $\psi$  given a unit change in the discrete variable  $X_i$ , while  $w_{ij}$  represents the frequency at which the change occurs. For the system given by (1),  $w_{ij} = k_{ij}X_i$ . To obtain the rate of the change of the expected value of  $X_i$ ,  $\frac{d}{dt}E[X_i]$ , we let  $\psi(X_i) = X_i$  in (2). Similarly, to obtain  $\frac{d}{dt}E[X_i^2]$ , we let  $\psi(X_i) = X_i^2$ .

*Example 1*: Consider the case when  $M = 2$ , *i.e.*,



where robots executing task 1,  $X_1$ , transition to task 2,  $X_2$ , and vice versa with rates  $k_{12}X_1$  and  $k_{21}X_2$  respectively. The first and second moment dynamics for  $X_1$  are given by

$$\begin{aligned} \frac{d}{dt}E[X_1] &= E\left[\left((X_1 + 1) - X_1\right)k_{21}X_2 + \left((X_1 - 1) - X_1\right)k_{12}X_1\right] \\ &= k_{21}E[X_2] - k_{12}E[X_1] \quad \text{and} \\ \frac{d}{dt}E[X_1^2] &= E\left[\left((X_1 + 1)^2 - X_1^2\right)k_{21}X_2 + \left((X_1 - 1)^2 - X_1^2\right)k_{12}X_1\right] \\ &= -2k_{12}E[X_1^2] + 2k_{21}E[X_1X_2] + k_{21}E[X_2] + k_{12}E[X_1]. \end{aligned}$$

When the  $w_{ij}$ 's are linear with respect to the system state  $\mathbf{X}$ , the moment equations are closed. This means that the time derivative for the first moment of  $X_i$ ,  $\frac{d}{dt}E[X_i]$ , is only dependent on the first moments of  $X_i$  for  $i = 1, \dots, M$ , *i.e.*,  $E[X_1], \dots, E[X_M]$ , the second moments are dependent on the first and second moments, and so on and so forth. This is important because when the moment equations are closed, the

moment dynamics can be expressed as a linear matrix equation as follows:

$$\frac{d}{dt} \begin{bmatrix} E[X_1] \\ E[X_2] \\ E[X_1 X_1] \\ E[X_2 X_2] \\ E[X_1 X_2] \end{bmatrix} = \begin{bmatrix} -k_{12} & k_{21} & 0 & 0 & 0 \\ k_{12} & -k_{21} & 0 & 0 & 0 \\ k_{12} & k_{21} & -2k_{12} & 0 & 2k_{21} \\ k_{12} & k_{21} & 0 & -2k_{21} & 2k_{12} \\ -k_{12} & -k_{21} & k_{12} & k_{21} & -k_{21} - k_{12} \end{bmatrix} \begin{bmatrix} E[X_1] \\ E[X_2] \\ E[X_1 X_1] \\ E[X_2 X_2] \\ E[X_1 X_2] \end{bmatrix}. \quad (3)$$

The steady state solution to the above equation lies in the nullspace of the coefficient matrix. The solution is unique if we consider the conservation constraint  $X_1 + X_2 = N$ , *i.e.*, the number of robots within the system is constant. For this example, the steady state solution is a binomial distribution with the probability for robots to be at site 1 (or executing task 1) given by  $p_1 = k_{21}(k_{12} + k_{21})^{-1}$ , and the mean and variance given by,  $E[X] = Np_1$  and  $E[(X - E[X])^2] = Np_1(1 - p_1)$  respectively.

In general, the ensemble moment dynamics for the system with  $M$  tasks/sites is given

$$\begin{aligned} \frac{d}{dt} E[X] &= \mathbf{K}E[X] \\ \frac{d}{dt} E[XX^T] &= \mathbf{K}E[XX^T] + E[XX^T]\mathbf{K}^T + \Gamma(\alpha, E[X]) \end{aligned} \quad (4)$$

where  $[\mathbf{K}]_{ij} = k_{ji}$  and  $[\mathbf{K}]_{ii} = -\sum_{(i,j) \in \mathcal{E}} k_{ij}$ . It is important to note that  $\mathbf{K}$  is a Markov process matrix and thus is negative semidefinite. This coupled with the conservation constraint  $\sum_i X_i = N$  leads to exponential stability of the system given by (4) [9, 13]. Each entry in the matrix of second moments is determined from the moment closure methods shown above where the entries of  $\Gamma(\alpha, E[X])$  are all linear with respect to the  $k_{ij}$ 's and the means  $E[X]$ . For the two state example given by equation (3),  $\Gamma(\alpha, E[X])$  is defined as

$$\Gamma(\alpha, E[X]) = \begin{bmatrix} k_{12}E[X_1] + k_{21}E[X_2] & -k_{12}E[X_1] - k_{21}E[X_2] \\ -k_{12}E[X_1] - k_{21}E[X_2] & k_{12}E[X_1] + k_{21}E[X_2] \end{bmatrix}.$$

Furthermore, the  $k_{ij}$ 's can be chosen to enable a team of robots to autonomously maintain some desired mean steady-state distribution of the team across the various tasks/sites [9, 12, 1]. In essence, the  $k_{ij}$ 's translate into a set of stochastic guard conditions for the single robot controllers. The result is a set of decentralized agent-level control policies that allow the team to maintain the steady-state mean of the ensemble distribution. Different from previous work, the focus of this paper is to use the ensemble moment dynamics to synthesize distributed control strategies to enable the team to maintain *both the mean and the variance* of the robot team distribution across the various tasks/sites. We describe the approach in the following sections.

### III. ENSEMBLE CONTROLLER DESIGN

As shown with equation (1), the rate in which agents in state  $X_i$  transition to  $X_j$  depends on the population in state  $X_i$ . As such, the more agents in state  $X_i$ , the faster they transition to  $X_j$ . However, Klavins recently showed that if we allow for both positive and negative transition rates, it is possible to shape both the mean and the variance of the ensemble distribution [13]. In other words, by introducing a negative feedback rate, it is possible to slow the population growth at a given state and thus affect the population variance in that state.

#### A. Controller Synthesis

Consider the following single reaction  $X_1 \xrightarrow{\alpha_{12}} X_2$  with the corresponding moment equation for  $X_1$  given by  $\frac{d}{dt} E[X_1] = -\alpha_{12}E[X_1]$ . If we add a negative feedback of the form  $u = -\beta X_2$  such that the ‘‘closed-loop’’ reaction becomes  $X_1 \xrightarrow{\alpha_{12} - \beta X_2} X_2$ , then the moment dynamics with state feedback is given by  $E[X_1] = -\alpha_{12}E[X_1] + \beta E[X_1 X_2]$  will depend on the covariant moment  $E[X_1 X_2]$ . Such a feedback control law breaks the linearity of the moment equations because the moment dynamics are not closed.

To ensure that the moment dynamics with state feedback remain closed, consider the following feedback controller

$$u = \beta \frac{X_2}{X_1}. \quad (5)$$

The reaction with state feedback becomes  $X_1 \xrightarrow{\alpha_{12} - \beta \frac{X_2}{X_1}} X_2$  where (5) can be seen as a form of linearizing feedback control that inhibits transitions from  $X_1$  to  $X_2$  as  $X_2$  becomes larger than  $X_1$ . For the two state system described in Example 1, the closed-loop reactions become

$$X_1 \xrightleftharpoons[\alpha_{21} - \beta_{21} \frac{X_1}{X_2}]{\alpha_{12} - \beta_{12} \frac{X_2}{X_1}} X_2 \quad (6)$$

and the corresponding closed-loop first and second moment dynamics for  $X_1$  become

$$\begin{aligned} \frac{d}{dt} E[X_1] &= (\alpha_{21} + \beta_{12})E[X_2] - (\alpha_{12} + \beta_{21})E[X_1], \\ \frac{d}{dt} E[X_1^2] &= (\alpha_{21} - \beta_{12})E[X_2] + (\alpha_{12} - \beta_{21})E[X_1] \\ &\quad + 2(\alpha_{21} + \beta_{21})E[X_1 X_2] - 2(\alpha_{12} + \beta_{12})E[X_1^2]. \end{aligned} \quad (7)$$

where the steady-state values of  $E[X_1]$  and  $E[X_1^2]$  can be independently set by adjusting parameters  $\alpha$  and  $\beta$ .

In general, for the  $M$  state system described by (4), we propose the following ensemble feedback controller

$$\mathbf{u} = -\mathbf{K}_\beta E[X] \quad \mathbf{K}_\beta^{ij} = \begin{cases} \beta_{ji} & \forall (i, j) \in \mathcal{E} \\ -\sum_{(i,j) \in \mathcal{E}} \beta_{ji} & \forall i = j \\ 0 & \text{otherwise} \end{cases}, \quad (8)$$

resulting with the following closed-loop moment dynamics

$$\begin{aligned} \frac{d}{dt} E[X] &= (\mathbf{K}_\alpha + \mathbf{K}_\beta)E[X] \\ \frac{d}{dt} E[XX^T] &= (\mathbf{K}_\alpha + \mathbf{K}_\beta)E[XX^T] + E[XX^T](\mathbf{K}_\alpha + \mathbf{K}_\beta)^T \\ &\quad + \Gamma(\alpha, \beta, E[X]). \end{aligned} \quad (9)$$

The above equations are obtained by simply substituting  $k_{ij} = \alpha_{ij} - \beta_{ij} \frac{X_j}{X_i}$  in the reactions given by (1) and applying the extended generator to  $\psi(X_i) = X_i$ .

#### B. Analysis

In this section, we show the stability of the ensemble feedback controller.

*Theorem 1:* The first moment dynamics of the system with ensemble feedback strategy given by (9) is stable.

*Proof:* The first moment dynamics for the system with ensemble feedback is given by (9)

$$\frac{d}{dt}E[X] = \mathbf{K}_\alpha E[X] + \mathbf{K}_\beta E[X]$$

Since both  $\mathbf{K}_\alpha$  and  $\mathbf{K}_\beta$  are Markov process matrices, they are negative semidefinite, each with a zero eigenvalue of multiplicity one. Furthermore, the eigenvector associated with the zero eigenvalue is the vector  $\mathbf{1}$  such that  $\mathbf{1}^T \mathbf{K}_{(\cdot)} = 0$ . The sum of two negative semidefinite matrices is still negative semidefinite and thus the first moment dynamics of the closed-loop system given by (9) is stable. ■

It is important to note that the rate of population exchange in the model allows for backwards flow when  $\beta_{ij}X_j > \alpha_{ij}X_i$ . In the systems that we are considering, we restrict this rate to be greater than or equal to zero. A rate of zero implies that no robots are executing that transition. This case requires a bit more work to show stability of the ensemble feedback strategy if only due to the saturation of the control inputs required to ensure  $\beta_{ij}X_j \leq \alpha_{ij}X_i$ .

#### IV. DISTRIBUTED IMPLEMENTATION

In this section we present the distributed implementation of the proposed ensemble feedback strategy given by equation (8) in the cases when robots have 1) full and infinite range communication and 2) local and finite range communication.

##### A. Full Communication

The feedback strategy (8) gives robots in state  $X_i$  the ability to set their own state transition rates to be independent from the number of robots in  $X_i$ . This, however, requires robots in  $X_i$  to know how many robots are at adjacent sites, *i.e.*, robots in  $X_j$  where  $e_{ij} \in \mathcal{E}$ . We begin with the assumption that individual robots have full knowledge of the ensemble states  $\mathbf{X}(t) = [X_1(t), \dots, X_M(t)]$ . In practice, this can be achieved by endowing each task site the ability to track the number of robots at the site and the ability to communicate with adjacent task sites. To obtain timely estimates of the ensemble states, robots would only need to communicate with its current task site.

Algorithm 1 describes the implementation of the individual robot controller shown in Fig. 1(a) where  $\mathcal{U}(0,1)$  denotes a continuous uniform distribution between 0 and 1. The algorithm consists of two parts: Lines 1:14 – where the robot executes task  $i$  and determines the time it stays with task  $i$ ,  $T_i$ , from the set of  $k_{ij}$ 's; and Lines 16:19 – where the robot determines the next task to execute after completing task  $i$ . Lines 16:19 in Algorithm 1 is necessary for the general case when task  $i$  can be followed by multiple tasks. To determine  $T_i$ , robots first calculate the sum of all the exit rates,  $\sum_{\forall i} k_{ij}$ , from  $i$  (Line 3). This total rate represents the rate a single robot leaves task  $i$  for  $j$  and is used to calculate the time a robot spends executing task  $i$ , *i.e.*,  $T_i$ . If a new robot arrives at task/site  $i$  or an existing one leaves, the robot recalculates  $T_i$ . The feedback controller given by equation (8) is implemented by assigning  $k_{ij} = \alpha_{ij} - \beta_{ij} \frac{X_j}{X_i}$  in Line 3 for all  $i, j$  pairs and

$X_i, X_j$  is obtained through local communication between the robot and site  $i$ .

We note that from the ensemble view, the parallel execution of Algorithm 1 by the team of  $N$  robots is akin to the parallel distribution of  $N$  stochastic simulation algorithms first proposed by Gillespie in [6]. Since the times a robot spends at a task/site are all exponentially distributed and memoryless, the process of gaining and losing robots at the site is a birth-death process which allows for the constant resetting of  $T_i$ .

---

##### Algorithm 1 Robot Controller w/ Full Communication

---

```

1: initialize  $robotState = U_i$ 
2: initialize all  $\alpha_{ij}$ s and  $\beta_{ij}$ s
3:  $k_{\Sigma i} = \sum_{\forall (i,j) \in \mathcal{E}} k_{ij}$ 
4:  $T_i = \sum_{\forall (i,j) \in \mathcal{E}} \frac{1}{k_{\Sigma i}} \ln \frac{1}{\mathcal{U}(0,1)}$ 
5: repeat
6:   execute  $U_i$ 
7:   if a new robot arrives at site  $i$  then
8:      $X_i = X_i + 1$ 
9:     goto 1
10:  end if
11:  if a robot leaves site  $i$  then
12:     $X_i = X_i - 1$ 
13:    goto 1
14:  end if
15: until  $t \leq T_i$ 
16:
17:  $j = \text{sample}[\frac{k_{i1}}{\sum_{\forall (i,j) \in \mathcal{E}} k_{ij}}, \dots, \frac{k_{ij}}{\sum_{\forall (i,j) \in \mathcal{E}} k_{ij}}, \dots]$ 
18:  $robotState = U_j$ 
19:  $X_j = X_j + 1$ 
20: goto 1

```

---

*Example 2:* Consider the example when  $M = 3$ . The equivalent chemical reaction network representation of the ensemble model is shown in Fig. 2 and the closed-loop first and second moment dynamics are given by equation (10). The values for the  $\alpha_{ij}$ 's and  $\beta_{ij}$ 's were chosen to achieve the desired population distribution mean of  $E[X] = [10, 15, 5]$  and variance of  $Var[X] = [4, 4, 2]$  at each site. These values are shown in Table I. By adjusting the ratios of the  $\alpha_{ij}$ 's and  $\beta_{ij}$ 's, we were able to maintain the same mean behavior while reducing the on-site population variance and simultaneously speeding up the closed-loop system's convergence rate. Fig. 3 shows the distribution of the ensemble at each of the three sites with and without the ensemble feedback strategy given by equation (8).

##### B. Local Communication

In practice, not only is it unreasonable to assume full and perfect communication among the robots, it is often unreasonable to assume full and perfect communication between sites. This is especially true when sites are distributed across vast geographic regions or in situations where long-range communication is difficult/impossible, *e.g.*, underground/underwater environments. In this section, we present a decentralized implementation of the proposed ensemble feedback control

$$\frac{d}{dt} \begin{bmatrix} E[X_1] \\ E[X_2] \\ E[X_3] \\ E[X_1^2] \\ E[X_2^2] \\ E[X_3^2] \\ E[X_1 X_2] \\ E[X_1 X_3] \\ E[X_2 X_3] \end{bmatrix} = \begin{bmatrix} -\alpha_{12} - \alpha_{13} - \beta_{21} & \alpha_{21} + \beta_{12} & \beta_{13} & 0 & 0 & 0 & 0 & 0 & 0 \\ \alpha_{12} + \beta_{21} & -\alpha_{21} - \beta_{12} - \beta_{32} & \alpha_{32} & 0 & 0 & 0 & 0 & 0 & 0 \\ \alpha_{13} & \beta_{32} & -\alpha_{32} - \beta_{13} & 0 & 0 & 0 & 0 & 0 & 0 \\ \alpha_{12} + \alpha_{13} - \beta_{21} & \alpha_{21} - \beta_{12} & -\beta_{13} & -2\alpha_{12} - 2\alpha_{13} - 2\beta_{21} & 0 & 0 & 2\alpha_{21} + 2\beta_{12} & 2\beta_{13} & 0 \\ \alpha_{12} - \beta_{21} & \alpha_{21} - \beta_{12} - \beta_{32} & \alpha_{32} & 0 & 0 & -2\alpha_{21} - 2\beta_{12} - 2\beta_{32} & 2\alpha_{12} + 2\beta_{21} & 0 & 2\alpha_{32} \\ \alpha_{13} & -\beta_{32} & \alpha_{32} - \beta_{13} & 0 & 0 & -2\alpha_{32} - 2\beta_{13} & 0 & 2\alpha_{13} & 2\beta_{32} \\ -\alpha_{13} & 0 & \beta_{13} & \alpha_{13} & 0 & 0 & \beta_{13} & -\alpha_{12} - \alpha_{13} - \alpha_{21} & \alpha_{32} \\ 0 & \beta_{32} & -\alpha_{32} & 0 & \alpha_{12} + \beta_{21} & \alpha_{21} + \beta_{12} & 0 & -\beta_{12} - \beta_{21} - \beta_{32} & \beta_{13} \\ 0 & \beta_{32} & -\alpha_{32} & 0 & \beta_{32} & \alpha_{32} & \alpha_{13} & \alpha_{12} + \beta_{21} & -\alpha_{21} - \alpha_{32} \\ & & & & & & & & -\beta_{12} - \beta_{13} - \beta_{32} \end{bmatrix} \begin{bmatrix} E[X_1] \\ E[X_2] \\ E[X_3] \\ E[X_1^2] \\ E[X_2^2] \\ E[X_3^2] \\ E[X_1 X_2] \\ E[X_1 X_3] \\ E[X_2 X_3] \end{bmatrix} \quad (10)$$

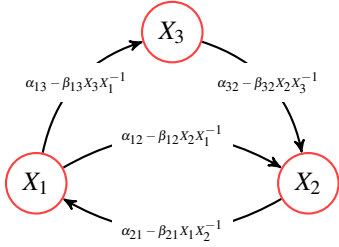


Fig. 2: A chemical reaction network representation of an example 3-site surveillance problem where  $X_i$  denotes the number of robots at site  $i$  executing surveillance task  $U_i$ .

Rates	$\alpha_{12}$	$\alpha_{13}$	$\alpha_{21}$	$\alpha_{32}$	$\beta_{12}$	$\beta_{21}$	$\beta_{13}$	$\beta_{21}$
w/o (8)	1.66	1.97	2.41	3.94	0	0	0	0
w/ (8)	2.93	2.52	2.82	6.98	0.85	1.01	0.60	1.01

TABLE I: Values of  $\alpha_{ij}$ 's and  $\beta_{ij}$ 's for the system given by equation (10) with and without the ensemble feedback strategy given by equation (8). These values were chosen such that  $E[X] = [10, 15, 5]$  and  $Var[X] = [4, 4, 2]$  with the ensemble feedback strategy.

strategy given by equation (8) that relies solely on local inter-robot communication. We assume robots have finite communication ranges and can only communicate with other robots that are co-located at the same site and/or within each other's communication range. As robots move from one site to another and exchange information with other robots they encounter, each robot can construct their own estimates of the population levels at the various sites.

To more faithfully represent the underlying agent-based system, we expand the ensemble model to take into account the navigation controller executed by individual robots as they move from one task/site to another. Given the set of  $M$  tasks and task controllers  $\{U_1, \dots, U_M\}$ , let  $U_{ij}$  denote the navigation controller executed by a robot to travel from site  $i$  to  $j$ . Let  $\lambda_{ij}$  denote the mean per robot arrival rates at site  $j$  for robots traveling from sites  $i$ . In other words,  $1/\lambda_{ij}$  denote the expected travel time between sites  $i$  and  $j$  whose variability can be affected by the number of robots "on the road". Let  $Y_{ij}(t)$  denote the number of robots traveling between sites  $i$  and  $j$ . Similar to the  $X_i(t)$  variables which denote the number of robots at site  $i$ ,  $Y_{ij}(t)$  are discrete random variables. The

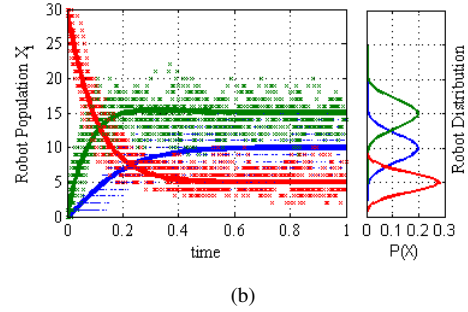
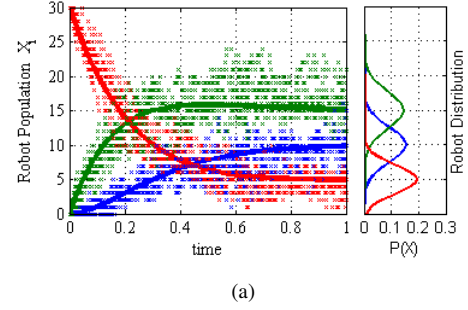


Fig. 3: Results of the system in Figure 2. These plots compare the behavior of the steady state distributions and the convergence rate of the system with and without ensemble feedback. Each left side plot shows the start up behavior from an initial condition of  $X = [0, 0, 30]$ . The solid lines denote the numerical solutions of the first moment dynamics and the data points are 10 representative stochastic simulation runs. The right side plots are the steady state distributions represented as normal distributions. Note how the system with ensemble feedback has faster convergence and small variance on its populations.

single robot controller and corresponding ensemble model is shown in Fig. 4(a).

Applying (2) with  $\psi(X_i) = X_i$ , we obtain the following first moment dynamics

$$\begin{aligned} \frac{d}{dt} E[X_i] &= \sum_{\forall j \rightarrow i} \lambda_{ji} E[Y_{ji}] - \sum_{\forall i \rightarrow j} k_{ij} E[X_i], \\ \frac{d}{dt} E[Y_{ij}] &= k_{ij} E[X_i] - \lambda_{ij} E[Y_{ij}] \end{aligned} \quad (11)$$

for all  $i, j = 1, \dots, M$ . Due to space considerations, we omit the equations for the second moment dynamics. Similar to previous examples, we employ the feedback control strategy

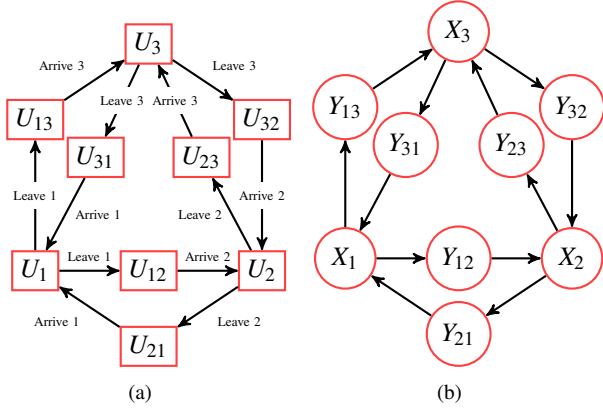


Fig. 4: (a) Robot controller for the 3-site surveillance problem with navigation.  $U_i$  denotes task controllers at site  $i$  and  $U_{ij}$  denotes navigation controllers between sites  $i$  and  $j$ . (b) The corresponding ensemble model.

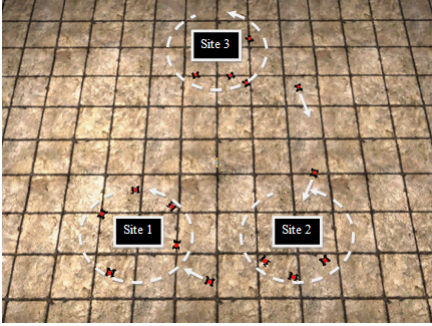


Fig. 5: A still of the 15 robot USAR simulation. While executing the survey task, the robots would circle the sites.

given by (8). The moment dynamics with state feedback can be obtained by substituting  $k_{ij} = \alpha_{ij} - \beta_{ij}X_jX_i^{-1}$  into the above equations.

To achieve online estimation of the on-site robot population as well as the robot population at neighboring sites by the individual robots, robots exchange information with other robots that are co-located at the same site. Robots arriving from site  $i$  from task  $j$  delivers an estimate of  $X_j$  which we denote as  $\hat{X}_{ji}$  and refer to it as the estimated value of  $X_j$  at task  $i$ . Since the proposed ensemble feedback strategy relies on information of robot populations at adjacent sites, the decentralized implementation requires two way reactions between sites to achieve variance control.

Algorithm 2 presents the decentralized single robot controller with ensemble feedback. We note that Line 9 denotes the exchange of information between robots arriving at site  $i$  from  $j$  with robots located at  $i$  and Line 20 denotes the information that a robot leaving site  $i$  will carry to robots located at site  $j$ . We present our simulation results in the following section.

## Algorithm 2 Robot Controller w/ Local Communication

```

1: initialize  $robotState = U_i$ 
2: initialize all  $\alpha_{ij}$ 's and  $\beta_{ij}$ 's
3:  $k_{\Sigma i} = \sum_{\forall(i,j) \in \mathcal{E}} k_{ij}(\alpha_{ij}, \beta_{ij})$ 
4:  $T_i = \sum_{\forall(i,j) \in \mathcal{E}} \frac{1}{k_{\Sigma i}} \ln \frac{1}{\mathcal{W}(0,1)}$ 
5: repeat
6:   execute  $U_i$ 
7:   if a new robot arrives at site  $i$  then
8:      $X_i = X_i + 1$ 
9:      $\hat{X}_{ji} = \hat{X}_j$ 
10:    goto 1
11:  end if
12:  if a robot leaves site  $i$  then
13:     $X_i = X_i - 1$ 
14:    goto 1
15:  end if
16: until  $t \leq T_i$ 
17:
18:  $j = \text{sample}[\frac{k_{i1}}{\sum_{\forall(i,j) \in \mathcal{E}} k_{ij}}, \dots, \frac{k_{ij}}{\sum_{\forall(i,j) \in \mathcal{E}} k_{ij}}, \dots]$ 
19:  $robotState = U_{ij}$ 
20:  $\hat{X}_i = X_i - 1$ 
21: goto 1

```

## V. RESULTS

To verify the validity of our robot controllers presented in Algorithms 1 and 2, we employed a multi-level simulation strategy. At the top level are the *macro-continuous* simulations where the linear moment closure equations are numerically solved. At the intermediate level are the *macro-discrete* simulations which are conducted using the Stochastic Simulation Algorithm (SSA) which is mathematically equivalent to an agent-based simulation [7]. At the lowest level are the *micro-discrete* simulations which are agent-based simulations using a team of mSRV-1 robots in USARSim [17]. A frame from one of our micro-discrete simulations is shown in Fig. 5.

Simulations at all three levels were run for an ensemble of 15 robots for the three site example shown in Fig. 4. The macro-discrete simulations were ran for approximately 30,000 transitions to ensure the system has reached steady-state for the following three scenarios: (a) the system with no ensemble feedback, and robots executing (b) Algorithm 1 and (c) Algorithm 2. The macro-continuous and macro-discrete results are shown in Fig. 6(a)-6(c). The macro-continuous results are represented as normal distributions and shown by the dotted lines in the figures. The values for  $\alpha_{ij}$ 's and  $\beta_{ij}$ 's in the ensemble feedback strategy are shown in Table III and chosen such that  $E[\hat{X}_i] = 4$  for each site while minimizing  $Var[\hat{X}_i]$ . The values for the expected travel times are shown in Table II. The same parameters were employed for the micro-discrete simulations for the same three scenarios: (d) the system with no ensemble feedback, and robots executing (e) Algorithm 1 and (f) Algorithm 2. The results are shown in Fig. 6(d)-6(f). Each micro-discrete distribution is constructed from 5 separate steady-state simulations each with 2,000 transitions.



TABLE II: Expected travel times,  $\lambda_{ij}^{-1}$ , between tasks.

Intersite Paths	$\lambda_{12}^{-1}$	$\lambda_{13}^{-1}$	$\lambda_{21}^{-1}$	$\lambda_{23}^{-1}$	$\lambda_{31}^{-1}$	$\lambda_{32}^{-1}$
$E[T_{ij}]$ (sec)	14.0	18.0	14.0	18.0	18.0	18.0

The first and second moments of our simulation results are shown in Table III for each scenario.

## VI. DISCUSSION

The results shown in Fig. 6 demonstrate the ability of our ensemble feedback strategy to simultaneously affect the mean and the variance of the on-site robot populations. More interestingly, our fully decentralized implementation where robots rely solely on at-site communication to estimate the various population variables achieves similar performance predicted by the ensemble model (see Fig. 6(c) and 6(f)).

However, we note that in our example, the mean transition rates were much slower than the travel transitions between sites,  $\alpha_{ij}E[X_i] - \beta_{ij}E[X_j] < \lambda_{ij}Y_{ij}$ . This led to a system where the effects due to the travel delays were overshadowed by the delays introduced by robots spending time at a site. For systems with larger travel times and lower site transition rates, the ensemble may encounter some instability due to poor estimation of the site populations. To alleviate these effects, we could provide the robots the ability to estimate both  $X_j$  and  $Y_{ij}$ . This is not entirely unreasonable since estimation of  $Y_{ij}$  could be achieved as robots encounter other robots as they travel between sites.

While there is some disagreement between the population means and variances predicted by the linear ensemble model and the actual robot simulations, we believe that the main source of the model error is due to the minimum transition rate of the single reactions. Each reaction expression has a rate of zero when either the number of robots at the associated site is zero, or the difference  $\alpha_{ij}E[X_i] - \beta_{ij}E[X_j] < 0$ . Since all of the rates are one-way, they only govern robots leaving a site and thus the transition rate will go to zero if  $\alpha_{ij}E[X_i] - \beta_{ij}E[X_j] < 0$ . The linear approximation does not take into account this saturation effect.

## VII. CONCLUSION AND FUTURE WORK

In this work, we presented a method for synthesizing distributed ensemble feedback control strategies through the development and analysis of an appropriate macroscopic description of the ensemble dynamics. Moment closure techniques were used to derive the ensemble dynamics and through this analysis a linearizing ensemble feedback strategy was obtained. We presented a distributed implementation of the proposed ensemble feedback strategy which can be implemented on robots with limited communication range. The resultant agent-level control policies enabled the team to affect both the mean and the variance of the ensemble population across the various spatially distributed tasks.

There are many directions for future work. From our preliminary results, we believe it is possible to select the appropriate feedback gains to not only affect the mean and the variance

of the on-site populations, but also to improve the systems overall convergence rate. We would like to investigate different optimization strategies to achieve this in an automated fashion. Furthermore, as mentioned in the discussion, the relationship between the length of the travel times and time required for individual robots to achieve a stable estimate of the various population variables are interrelated. We are interested in studying the sensitivity of these ensemble feedback strategies in situations where the travel times may be large. Finally, we would like to determine the appropriate bounds on system and controller parameters to ensure stability of our distributed ensemble feedback control policies.

## REFERENCES

- [1] S. Berman, A. M. Halasz, M. A. Hsieh, and V. Kumar. Navigation-based optimization of stochastic strategies for allocating a robot swarm among multiple sites. In *Proc. 2008 IEEE Conf. on Decision & Control (CDC'08)*, pages 4376–4381, Cancun, Mexico, 2008. URL <http://dx.doi.org/10.1109/CDC.2008.4739482>.
- [2] T. S. Dahl, M. J. Mataric, and G. S. Sukhatme. A machine learning method for improving task allocation in distributed multi-robot transportation. In D. Braha, A. Minai, and Y. Bar-Yam, editors, *Understanding Complex Systems: Science Meets Technology*, pages 307–337. Springer, Berlin, Germany, June 2006.
- [3] M. B. Dias, R. M. Zlot, N. K., and A. Stentz. Market-based multirobot coordination: a survey and analysis. *Proc. of the IEEE*, 94(7):1257–1270, July 2006. URL <http://dx.doi.org/10.1109/JPROC.2006.876939>.
- [4] B. P. Gerkey and M. J. Mataric. Sold!: Auction methods for multi-robot control. *IEEE Trans. on Robotics & Automation*, 18(5):758–768, Oct 2002.
- [5] B. P. Gerkey and M. J. Mataric. A formal framework for the study of task allocation in multi-robot systems. *Int. J. of Robotics Research*, 23(9):939–954, September 2004.
- [6] D. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J. of Computational Physics*, 22(4):403–434, 1976.
- [7] D. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. of Physical Chemistry*, 81:2340–2361, 1977. URL <http://dx.doi.org/10.1021/j100540a008>.
- [8] M. Golfarelli, D. Maio, and S. Rizzi. Multi-agent path planning based on task-swap negotiation. In *Proc. 16th UK Planning & Scheduling SIG Workshop*, Durham, England, 1997. PlanSIG.
- [9] A. Halasz, M. A. Hsieh, S. Berman, and V. Kumar. Dynamic redistribution of a swarm of robots among multiple sites. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'07)*, pages 2320–2325, San Diego, CA, Oct.–Nov. 2007. URL <http://dx.doi.org/10.1109/IROS.2007.4399528>.
- [10] J. P. Hespanha. Moment closure for biochemical networks. In *Proc. of the 3rd Int. Symp. on Control, Communications and Signal Processing*, Mar.

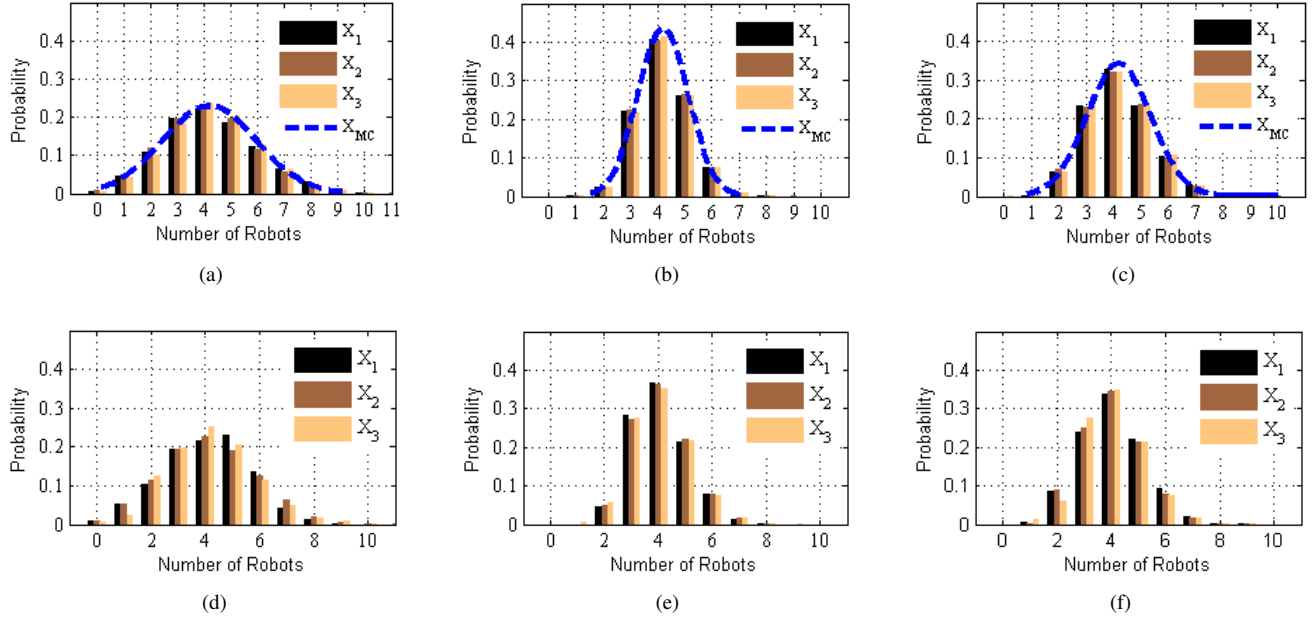


Fig. 6: Probability distribution of the robot ensemble at each site. Macro-continuous and macro-discrete simulation results for the system (a) w/o ensemble feedback, (b) w/ ensemble feedback (Algorithm 1), and (c) w/ ensemble feedback (Algorithm 2). Micro-discrete simulation results for the system (d) w/o ensemble feedback, (e) w/ ensemble feedback (Algorithm 1), and (f) w/ ensemble feedback (Algorithm 2).

	$\alpha_{ij}$	$\beta_{ij}$	$E[X_1]$	$E[X_2]$	$E[X_3]$	$Var[X_1]$	$Var[X_2]$	$Var[X_3]$
(a)	0.0056	0.0000	4.22	4.21	4.19	3.05	3.06	3.09
(d)	0.0056	0.0000	4.11	4.12	4.17	2.63	2.94	2.71
(b)	0.0178	0.0112	4.19	4.17	4.17	0.93	0.95	0.96
(e)	0.0178	0.0112	4.04	4.07	4.18	1.12	1.19	1.25
(c)	0.0178	0.0112	4.19	4.18	4.18	1.50	1.53	1.48
(f)	0.0178	0.0112	4.06	3.99	3.98	1.47	1.36	1.32

TABLE III: Values of  $\alpha_{ij}$  and  $\beta_{ij}$  for the system shown in Fig. 4(a) and resulting first and second moments.

2008. URL <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4537208&isnumber=4537177>.
- [11] K. Hosokawa, I. Shimoyama, and H. Miura. Dynamics of self assembling systems: Analogy with chemical kinetics. *Artificial Life*, 1(4):413–427, 1994.
- [12] M. A. Hsieh, A. H., S. Berman, and V. Kumar. Biologically inspired redistribution of a swarm of robots among multiple sites. *Swarm Intelligence*, December 2008. URL <http://dx.doi.org/10.1007/s11721-008-0019-z>.
- [13] E. Klavins. Proportional-integral control of stochastic gene regulatory networks. In *Proc. 2010 IEEE Conf. on Decision & Control (CDC'10)*, 2010.
- [14] K. Lerman, C. Jones, A. Galstyan, and M. J. Mataric. Analysis of dynamic task allocation in multi-robot systems. *Int. J. of Robotics Research*, 2006. URL <http://dx.doi.org/10.1177/0278364906063426>.
- [15] A. Martinoli, K. Easton, and W. Agassounon. Modeling of swarm robotic systems: a case study in collaborative distributed manipulation. *Int. J. of Robotics Research: Special Issue on Experimental Robotics*, 23(4-5):415–436, 2004. URL <http://dx.doi.org/10.1177/0278364904042197>.
- [16] N. Napp, S. Burden, and E. Klavins. Setpoint regulation for stochastically interacting robots. In *Robotics: Science and Systems V*. MIT Press, 2009. URL <http://dx.doi.org/10.1007/s10514-010-9203-2>.
- [17] USARSim. Unified system for automation and robot simulation. <http://usarsim.sourceforge.net>, 2007.