Adaptive Communication-Constrained Deployment of Unmanned Vehicle Systems

Jerome Le Ny, *Member, IEEE*, Alejandro Ribeiro, *Member, IEEE*, and George J. Pappas, *Fellow, IEEE*

Abstract—Cooperation between multiple autonomous vehicles requires inter-vehicle communication, which in many scenarios must be established over an ad-hoc wireless network. This paper proposes an optimization-based approach to the deployment of such mobile robotic networks. A primal-dual gradient descent algorithm jointly optimizes the steady-state positions of the robots based on the specification of a high-level task in the form of a potential field, and routes packets through the network to support the communication rates desired for the application. The motion planning and communication objectives are tightly coupled since the link capacities depend heavily on the relative distances between vehicles. The algorithm decomposes naturally into two components, one for position optimization and one for communication optimization, coupled via a set of Lagrange multipliers. Crucially and in contrast to previous work, our method can rely on on-line evaluation of the channel capacities during deployment instead of a prespecified model. In this case, a randomized sampling scheme along the trajectories allows the robots to implement the algorithm with minimal coordination overhead.

Index Terms—Mobile wireless network optimization, unmanned vehicle systems, robot motion planning, primal-dual optimization algorithms.

I. INTRODUCTION

UNMANNED Vehicle Systems (UVS) have become critical assets for intelligence, surveillance and reconnaissance missions [1], and could be used in the near future for environmental monitoring, search and rescue missions, intelligent distribution and transportation, or the exploration of dangerous indoor environments. UVS consist of a number of autonomous vehicles or mobile robots that can communicate with each other to enable cooperative behaviors [2], [3], and with base stations for complex data analysis and higher level control purposes. Thus reliable and efficient wireless communication is a key component to the successful operation of current and future UVS [1]. Except for the largest unmanned vehicles communicating with distant command centers via satellite links, the bulk of these systems must rely on line-of-sight communication [1]. In many situations, notably for military operations or disaster relief operations, a wireless communication infrastructure is initially absent and the robots need to form an ad-hoc network [4]. Hence with the currently

increasing density of vehicles simultaneously in operations, spectrum management and network optimization is becoming a major issue for UVS [1].

The communication quality requirements for UVS favor small inter-vehicle distances or high altitudes and are therefore in conflict with the operational requirements, since better sensing requires low altitude operations and better exploration requires spreading the network. Much work in mobile robotics has focused on trajectory planning and deployment under communication constraints, but generally by assuming very simplified connectivity constraints and communication models. In particular, the disc model enables the use of graph theoretical methods to account for connectivity constraints, see e.g. [5], [6]. However, such models are inadequate to optimize global performance metrics for the network, such as bandwidth specifications between distant terminals, or to exploit the spatial diversity in the inter-robot channels to adaptively route the traffic. Interferences are also not taken into account.

Recently however, more realistic wireless channel models have been used in robotics [7]–[9]. Firouzabadi and Martins [10] consider a network optimization problem including node placement and optimal power allocation, and solve it using geometric programming. Zavlanos et al. [11], [12] consider a problem similar to the one presented in this paper, including robot motion planning and packet routing. Other related communication-constrained deployment problems are treated in e.g. [13], [14], and some experimental studies can be found in [15]–[17]. However, virtually all proposed approaches rely on known, and often deterministic models of the channel gains. In this case, the resulting trajectory planning algorithms can operate "open-loop", without ever relying on wireless channel measurements. Unfortunately wireless channel modeling is notably difficult, especially for indoor environments [4], and the lack of robustness of the proposed methods to modeling errors has been discussed only in a few cases [7], [9]. Fink et al. [13], [18], [19] discuss a robotic network controller that includes a separate communication channel estimation component, but in general little is known on how to naturally integrate on-line wireless channel measurements in a feedback motion planner.

Contributions and Outline. As described in Section II, we consider generic communication-constrained *deployment problems*, where the goal is to move a mobile robotic network from an initial configuration to a desired steady-state configuration appropriate for a given task, without directly attempting to optimize its transient trajectory. This framework has a wide range of applications in robotics [20], including

formation control [21], coverage control [22], dynamic vehicle routing [23], or target tracking [24], and can also be used to continuously adapt the configuration in slowly varying environments. One of its main advantages is that it typically leads to distributed algorithms, which are more amenable to implementation in large networks. In Section III, we develop a primal-dual optimization algorithm that progressively drives the robots toward a satisfying spatial configuration for the task, while ensuring that the resulting network can support the desired communication flows between vehicles and with the base stations. This method only weakly couples the motion planning and communication optimization components, via a set of Lagrange multipliers. Moreover, an attractive feature of the algorithm is that it can be easily modified to plan the trajectories using channel measurements rather than prespecified channel models, as detailed in Section IV. The measurement-based algorithm relies in particular on a random sampling direction approach proposed in the stochastic optimization literature [25], [26], which requires little additional coordination between the robots. Formal convergence results for the algorithms are stated in Sections III-C and IV-B, and simulation results for three illustrative scenarios are discussed in Section V. Finally, we conclude in Section VI.

II. JOINT DEPLOYMENT AND COMMUNICATION OPTIMIZATION

A. Task Potential

Consider a mobile robotic network consisting of n robots evolving in a workspace $\mathsf{W} \subset \mathbb{R}^d$, with positions denoted $\mathbf{x} = [\mathbf{x}_1^T, \ldots, \mathbf{x}_n^T]^T \in \mathsf{W}^n$. We use boldface letters for vector quantities, and denote $[n] := \{1, \ldots, n\}$. For simplicity, we assume throughout the paper that W is compact and convex. Some of the robot positions may be fixed, so that these "robots" can also represent a fixed communication infrastructure composed of base stations, with which the mobile elements must maintain communication. We consider high-level motion planning problems where we neglect the dynamics of the robots, whose positions evolve in discrete time as

$$\mathbf{x}_i[k+1] = \mathcal{P}_{\mathsf{W}}(\mathbf{x}_i[k] + \mathbf{u}_i[k]), \quad \mathbf{x}_i[0] \in \mathsf{W}, \quad \forall i \in [n]. \quad (1)$$

Here $\mathcal{P}_{\mathsf{W}}$ denotes the projection on W used to keep the iterates within the workspace, and $\mathbf{u}_i[k]$ is the control input for robot $i \in [n]$ at period k, satisfying the velocity constraint $\|\mathbf{u}_i[k]\| \leq v_i$, where $\|\cdot\|$ denotes the Euclidean norm.

The quality of the deployment of the mobile robotic network is captured by a potential field $G(\mathbf{x})$ [21], [27], [28], whose minimum corresponds to a desired steady-state configuration $\mathbf{x}^*$ for the group. We call this function the *task potential*. The deployment problem for this task potential consists then in designing feedback control laws allowing the system to reach $\mathbf{x}^*$ starting from an initial configuration $\mathbf{x}[0]$. For example, a simple quadratic potential

$$G(\mathbf{x}) = \|\mathbf{x}_1 - \mathbf{q}^*\|^2 \quad (2)$$

can be used to bring robot 1 to a known target location $\mathbf{q}^*$. Diverging barrier potentials can be used for inter-robot and obstacle avoidance in non-convex workspaces [27], [28], and

other potential functions can force groups of robots to maintain certain formations [21]. Another example of a task potential for a group of robots is the *coverage control* task potential introduced by Cortés et al. [22]. To describe it, consider a scenario where some events of interest can occur in the workspace with their location distributed according to some probability measure $\mathbb{P}_Z$ on W. The coverage control problem consists in finding a steady-state configuration for the robotic network that minimizes the objective

$$G(\mathbf{x}) = \mathbb{E}_Z \left[\min_{i \in [n]} c(\|Z - \mathbf{x}_i\|) \right], \quad (3)$$

where $\mathbb{E}_Z$ is the expectation operator corresponding to $\mathbb{P}_Z$ and c is an increasing, continuously differentiable function. This objective measures the steady-state average "service" performance for the events, assuming that an event is serviced by the robot closest to its location, say robot i, at a cost $c(\|Z - \mathbf{x}_i\|)$. Note that for $c(x) = x^2$, the potential (3) is nothing but the quadratic vector quantization performance index [29], with the robot positions playing the role of the quantizers.

For a deployment problem with task potential G, a popular type of feedback controller takes a truncated gradient form [27]

$$\mathbf{u}(\mathbf{x}[k]) = \mathrm{sat}_{\mathbf{v}}(-\alpha_k \nabla G(\mathbf{x}[k])) \quad (4)$$

where α_k are prespecified stepsizes, $\mathbf{v} = [v_1, \ldots, v_n]$, and for $i \in [n]$

$$(\mathrm{sat}_{\mathbf{v}}(\mathbf{u}))_i = \begin{cases} \mathbf{u}_i, & \text{if } \|\mathbf{u}_i\| \leq v_i \\ v_i \frac{\mathbf{u}_i}{\|\mathbf{u}_i\|}, & \text{otherwise.} \end{cases}$$

The resulting gradient descent algorithm (1) only leads to the set of critical points of G in general. We follow this approach nonetheless, as global minimization of interesting task potentials for most multi-robot deployment problems, such as (3), is computationally intractable.

B. Wireless Networking

Because the deployed robotic network must also satisfy certain communication constraints, the gradient controller (4) needs to be modified. Indeed, path loss and interferences make the achievable wireless communication rates highly dependent on inter-robot distances. In this section we review some terminology and basic principles of wireless networking, in order to express these communication constraints quantitatively.

Each robot is equipped with a wireless terminal, and wishes to deliver communication packets to other robots and base stations for different application level flows, where a flow ϕ is associated to a given final destination $\mathrm{dest}(\phi)$. Note that several flows can have the same destination. The set of flows is denoted Φ. The amount of information for flow $\phi \in \Phi$ accepted at robot $i \neq \mathrm{dest}(\phi)$ at time k is denoted $a_i^\phi[k]$, with $a_i^\phi[k] \geq 0$. The amount of information for flow ϕ routed between robots i and j at time k is denoted $r_{ij}^\phi[k]$, with $r_{ij}^\phi[k] \geq 0$. Finally, the capacity of the link (i, j) at time k if the robotic network is in configuration $\mathbf{x}$ is denoted $c_{ij}[k; \mathbf{x}]$. This capacity is also nonnegative and in general random, determined at the physical layer by the fading state and

the chosen communication scheme, through coding and the allocation of transmission time-slots, frequencies and powers [4].

Let us illustrate in more details how the robot positions influence the channel capacities. Assume that a set $\mathcal{F}$ of frequency tones is available to communicate. For every pair $(i,j) \in [n]$, let $h_{ij}^f[k; \mathbf{x}_i, \mathbf{x}_j]$ denote the channel power gain at period k on frequency f, from terminal i at position $\mathbf{x}_i$ to terminal j at position $\mathbf{x}_j$. Following standard practice in wireless communications, we assume that $\{h_{ij}^f[k; \mathbf{x}_i, \mathbf{x}_j]\}_{f,ij}$ is the realization of a random vector $\mathbf{H}[k; \mathbf{x}]$ [4]. Channel gain models generally take the form

$$h_{ij}^f[k; \mathbf{x}_i, \mathbf{x}_j]|_{dB} := 10 \log_{10} h_{ij}^f[k; \mathbf{x}_i, \mathbf{x}_j]$$
$$= l^f(\mathbf{x}_i, \mathbf{x}_j) + Y_{ij}^f[k], \quad (5)$$

where $Y_{ij}^f[k]$ is a zero-mean random variable modeling fading and shadowing effects [4]. The function $l^f(\mathbf{x}_i, \mathbf{x}_j)$ models the deterministic path loss between positions $\mathbf{x}_i$ and $\mathbf{x}_j$. Let p_{ij}^f denote the power used by terminal i to communicate with terminal j over the frequency tone f. Many communication schemes result in link capacities that are functions of the signal to interference plus noise ratios (SINR)

$$\text{SINR}_{ij}^f[k; \mathbf{x}] = \frac{h_{ij}^f[k; \mathbf{x}_i, \mathbf{x}_j] p_{ij}^f}{\sigma_j + \frac{1}{S} \sum_{(l,m) \neq (i,j)} h_{lj}^f[k; \mathbf{x}_l, \mathbf{x}_j] p_{lm}^f}, \quad (6)$$

where σ_j denotes the noise power at receiver j and $1/S$ is the interference reduction due to signal processing, e.g., S is approximately equal to the processing gain in a CDMA system [4, chapter 14]. With capacity-achieving channel codes, we could have for example $c_{ij}[k; \mathbf{x}] = \sum_{f \in \mathcal{F}} \log \left(1 + \text{SINR}_{ij}^f[k; \mathbf{x}]\right)$.

Remark 1: Frequency and power allocation optimization can also be included in our framework, following [30]. These variables are not included below for clarity of exposition, our focus being on the role of controlled mobility on communication performance.

Coming back to the communication constraints, to ensure boundedness of the queues at all terminals, it is sufficient to ensure that the long-term average amount of information accepted at each terminal is less than the long-term average amount of information forwarded to other terminals [31], i.e.,

$$\lim_{T \to \infty} \frac{1}{T} \sum_{k=0}^{T-1} \left(a_i^\phi[k] + \sum_{j \neq i} r_{ji}^\phi[k]\right) \leq \lim_{T \to \infty} \frac{1}{T} \sum_{k=0}^{T-1} \sum_{j \neq i} r_{ij}^\phi[k]$$
$$\forall \phi, \forall i \neq \text{dest}(\phi), \quad (7)$$

assuming for now that the limits exist. We denote the long-term averages in the following by

$$a_i^\phi := \lim_{T \to \infty} \frac{1}{T} \sum_{k=0}^{T-1} a_i^\phi[k], \quad r_{ij}^\phi := \lim_{T \to \infty} \frac{1}{T} \sum_{k=0}^{T-1} r_{ij}^\phi[k], \quad i \neq j.$$

Moreover, the long-term average amount of information circulating on link (i,j) cannot exceed the long-term average capacity. We assume that for each fixed configuration $\mathbf{x}$, the capacity values $c_{ij}[k; \mathbf{x}]$ form a temporal sequence of identically and independently distributed (iid) random variables,

with mean $c_{ij}(\mathbf{x})$. By the strong law of large number, we have

$$c_{ij}(\mathbf{x}) = \mathbb{E}[c_{ij}[0; \mathbf{x}]] = \lim_{T \to \infty} \frac{1}{T} \sum_{k=0}^{T-1} c_{ij}[k; \mathbf{x}], \quad \text{a.s.} \quad (8)$$

In fact, the algorithm below works with general ergodic Markov chains instead of iid sequences, with more technical assumptions for convergence in Theorem 5. For a robotic network in a fixed configuration $\mathbf{x}$, we then have the constraint

$$\lim_{T \to \infty} \frac{1}{T} \sum_{k=0}^{T-1} \sum_{\phi \in \Phi} r_{ij}^\phi[k] \leq \lim_{T \to \infty} \frac{1}{T} \sum_{k=0}^{T-1} c_{ij}[k; \mathbf{x}], \forall i \neq j. \quad (9)$$

With the introduced notation, we can rewrite the inequality constraints (7) and (9) simply as $a_i^\phi \leq \sum_{j \neq i} r_{ij}^\phi - r_{ji}^\phi$ and $\sum_{\phi \in \Phi} r_{ij}^\phi \leq c_{ij}(\mathbf{x})$. We also adopt the convention $r_{\text{dest}(\phi)j}^\phi = 0$ for all j, a natural requirement. Finally, we can have rate constraints of the form $a_{i,\min}^\phi \leq a_i^\phi \leq a_{i,\max}^\phi$ for $i \neq \text{dest}(\phi)$ and $r_{ij,\min}^\phi \leq r_{ij}^\phi \leq r_{ij,\max}^\phi$ for $i \neq j$, with $a_{i,\min}^\phi, r_{ij,\min}^\phi = 0$ and $a_{i,\max}^\phi, r_{ij,\max}^\phi = +\infty$ possible values. Any set of average rates $\mathbf{a} := \{a_i^\phi\}_{i,\phi}, \mathbf{r} := \{r_{ij}^\phi\}_{i \neq j,\phi}$ and final configuration $\mathbf{x} \in \mathsf{W}^n$ such that the constraints above are satisfied is said to be feasible. We would like to select, among these feasible points, one that is (at least locally) optimal according to a given criterion, including the task potential for the configuration component.

C. Joint Optimization Problem

We introduce concave utilities $U_i^\phi(a_i^\phi)$ to value the average admission rates a_i^ϕ, and convex costs $V_{ij}^\phi(r_{ij}^\phi)$ for establishing communication links. The optimal configuration and wireless network parameters are then defined as the solution of the following optimization problem

$$\min_{\mathbf{x}, \mathbf{a}, \mathbf{r}} \quad G(\mathbf{x}) - \sum_{\phi \in \Phi} \sum_{i \neq \text{dest}(\phi)} U_i^\phi(a_i^\phi) + \sum_{\phi \in \Phi} \sum_{i \neq j} V_{ij}^\phi(r_{ij}^\phi) \quad (10)$$

$$\text{s.t.} \quad a_i^\phi \leq \sum_{j \neq i} r_{ij}^\phi - r_{ji}^\phi, \quad \forall \phi, \forall i \neq \text{dest}(\phi) \quad (11)$$

$$\sum_{\phi \in \Phi} r_{ij}^\phi \leq c_{ij}(\mathbf{x}), \quad \forall i \neq j, \quad (12)$$

$$a_{i,\min}^\phi \leq a_i^\phi \leq a_{i,\max}^\phi, \quad \forall \phi, \forall i \neq \text{dest}(\phi), \quad (13)$$

$$r_{ij,\min}^\phi \leq r_{ij}^\phi \leq r_{ij,\max}^\phi, \quad \forall i \neq j. \quad (14)$$

For a fixed configuration $\mathbf{x}$, the problem (10)-(14) is convex in the communication rates $\mathbf{a}, \mathbf{r}$. However, the presence of the configuration vector $\mathbf{x}$ makes the problem non-convex in general, because most useful multi-robot task potentials G are not convex and the mean capacities $c_{ij}(\mathbf{x})$ in (12) are not concave functions of $\mathbf{x}$. Still, following standard practice in robotics [21]–[23], [27], one can at least look for a locally optimal solution to the constrained problem (10)-(14). Moreover, by using a gradient-based algorithm similar in spirit to (4), we can obtain a feedback controller driving the robotic network progressively toward this locally optimal configuration. The next section introduces a primal-dual optimization algorithm for this purpose.

III. PRIMAL-DUAL ALGORITHM

We assume in this section that the average channel capacities $c_{ij}(\mathbf{x})$ in (12) are known functions. Problem (10)-(14) is then a deterministic optimization problem, for which many nonlinear programming techniques are available [32]. Here we concentrate on a primal-dual augmented Lagrangian algorithm [32]. Section III-B motivates this choice.

Define the dual variables $\lambda_i^\phi \geq 0, i \neq$ dest ϕ, and $\mu_{ij} \geq 0, i \neq j$, associated with (11) and (12) respectively. We gather the primal and dual variables in the vectors $\mathbf{y} = (\mathbf{x}, \mathbf{a}, \mathbf{r})$ and $\boldsymbol{\xi} = (\boldsymbol{\lambda}, \boldsymbol{\mu})$ respectively. The region defined by W and the box constraints (13), (14) for the primal variables is denoted Y. Next, let $F(\mathbf{y})$ denote the objective function (10), and denote the functions appearing in the constraints (11), (12) as

$$g_i^\phi(a_i^\phi, \mathbf{r}_{i\cdot}^\phi, \mathbf{r}_{\cdot i}^\phi) = a_i^\phi - \sum_{j \neq i}\left(r_{ij}^\phi - r_{ji}^\phi\right), \quad \forall \phi, \forall i \neq \text{dest}(\phi)$$

$$h_{ij}(\mathbf{x}, \mathbf{r}_{ij}) = \sum_{\phi \in \Phi} r_{ij}^\phi - c_{ij}(\mathbf{x}), \quad \forall i \neq j,$$

with $\mathbf{r}_{ij} = \{r_{ij}^\phi\}_{\phi \in \Phi}$, $\mathbf{r}_{\cdot i}^\phi = \{r_{ji}^\phi\}_{j \in [n]}$, and $\mathbf{r}_{i\cdot}^\phi = \{r_{ij}^\phi\}_{j \in [n]}$. We can always rewrite an inequality constraint $K(\mathbf{y}) \leq 0$ as an equality constraint $K(\mathbf{y}) + z^2 = 0$, with z a slack variable. Doing this for (11), (12) leads to the definition of the augmented Lagrangian [32, Section 4.2] for the equivalent equality-constrained problem

$$\hat{\mathcal{L}}_\rho(\mathbf{y}, \mathbf{z}, \boldsymbol{\xi}) = F(\mathbf{y}) + \sum_{\substack{\phi \in \Phi \\ i \neq \text{dest}(\phi)}} \left\{ \lambda_i^\phi \left(g_i^\phi(a_i^\phi, \mathbf{r}_{i\cdot}^\phi, \mathbf{r}_{\cdot i}^\phi) + (z_i^\phi)^2\right) \right.$$
$$\left. + \frac{\rho}{2}|g_i^\phi(a_i^\phi, \mathbf{r}_{i\cdot}^\phi, \mathbf{r}_{\cdot i}^\phi) + (z_i^\phi)^2|^2 \right\}$$
$$+ \sum_{i=1}^n \sum_{j \neq i} \left\{ \mu_{ij}\left(h_{ij}(\mathbf{x}, \mathbf{r}_{ij}) + z_{ij}^2\right) + \frac{\rho}{2}|h_{ij}(\mathbf{x}, \mathbf{r}_{ij}) + z_{ij}^2|^2 \right\},$$

where $\rho > 0$ is a penalization parameter, and the variables z_i^ϕ and z_{ij} are slack variables. Partial minimization with respect to the slack variables, as detailed in [32, p. 406], allows us to work with the following simpler version of the augmented Lagrangian function, which depends only on the original primal and dual variables

$$\mathcal{L}_\rho(\mathbf{y}, \boldsymbol{\xi}) = F(\mathbf{y}) + \qquad\qquad\qquad (15)$$
$$\frac{1}{2\rho} \sum_{\phi \in \Phi} \sum_{i \neq \text{dest}(\phi)} \left\{ (\max\{0, \lambda_i^\phi + \rho g_i^\phi(a_i^\phi, \mathbf{r}_{i\cdot}^\phi, \mathbf{r}_{\cdot i}^\phi)\})^2 - (\lambda_i^\phi)^2 \right\}$$
$$+ \frac{1}{2\rho} \sum_{i=1}^n \sum_{j \neq i} \left\{ (\max\{0, \mu_{ij} + \rho h_{ij}(\mathbf{x}, \mathbf{r}_{ij})\})^2 - \mu_{ij}^2 \right\}.$$

Many optimization algorithms aim at computing a Karush-Kuhn-Tucker (KKT) point $(\mathbf{y}^*, \boldsymbol{\xi}^*)$ where the necessary optimality conditions for (10)-(14) are satisfied [32]. Consider a first-order primal-dual algorithm of the form

$$\mathbf{y}[k+1] = \mathcal{P}_Y\left(\mathbf{y}[k] - \alpha_k \nabla_\mathbf{y}\mathcal{L}_\rho(\mathbf{y}[k], \boldsymbol{\xi}[k])\right) \qquad (16)$$

$$\lambda_i^\phi[k+1] = \left[\lambda_i^\phi[k] + \beta_k g_i^\phi(a_i^\phi[k], \mathbf{r}_{i\cdot}^\phi[k], \mathbf{r}_{\cdot i}^\phi[k])\right]_0^{\lambda_{i,\max}^\phi} \qquad (17)$$

$$\mu_{ij}[k+1] = [\mu_{ij}[k] + \beta_k h_{ij}(\mathbf{x}[k], \mathbf{r}_{ij}[k])]_0^{\mu_{ij,\max}}, \qquad (18)$$

where α_k, β_k are prespecified stepsizes chosen as discussed in Section III-C, $\mathcal{P}_Y$ denotes the projection on the set Y, and $\nabla_\mathbf{y}$ denotes the vector of derivatives with respect to the primal variables. We use the notation $[x]_l^u := \max\{l, \min\{u, x\}\}$ to project $x \in \mathbb{R}$ on the interval $[l, u]$. The upper bounds $\lambda_{i,\max}^\phi$ and $\mu_{ij,\max}$ need to be sufficiently large so that the resulting box region

$$\Xi := \{(\boldsymbol{\lambda}, \boldsymbol{\mu}) | 0 \leq \lambda_i^\phi < \lambda_{i,\max}^\phi, \forall i \neq \text{dest } \phi,$$
$$0 \leq \mu_{ij} < \mu_{ij,\max}, \forall i \neq j\} \qquad (19)$$

contains the desired Lagrange multiplier $\boldsymbol{\xi}^*$. They can be set to $+\infty$ if no such region estimate is known, but otherwise can be used to significantly reduce oscillations in the primal variable trajectories, see [33, p. 181]. Upper bounds on Lagrange multipliers can be obtained for example from duality arguments [34, p. 6377].

A. Explicit Form and Considerations about Distributed Implementations

This section describes the nice structure given to the generic primal variable update equation (16) by the separable form of the augmented Lagrangian (15). First, for a differentiable function K and parameters λ, ρ, we have $\nabla_\mathbf{y}[(\max\{0, \lambda + \rho K(\mathbf{y})\})^2] = 2\rho \max\{0, \lambda + \rho K(\mathbf{y})\}\nabla_\mathbf{y}K(\mathbf{y})$. We can thus rewrite (16) explicitly as

$$\forall i \in [n], \ \mathbf{x}_i[k+1] = \mathcal{P}_W\left(\mathbf{x}_i[k] + \text{sat}_\mathbf{v}\left[-\alpha_k\left(\frac{\partial G}{\partial \mathbf{x}_i}(\mathbf{x}[k]) - \right.\right.\right.$$
$$\left.\left.\left. \sum_{l,m \neq l} \max\{0, \mu_{lm}[k] + \rho h_{lm}(\mathbf{x}[k], \mathbf{r}_{lm}[k])\}\frac{\partial c_{lm}}{\partial \mathbf{x}_i}(\mathbf{x}[k])\right)\right]\right),$$
$$(20)$$

$$a_i^\phi[k+1] = \left[a_i^\phi[k] - \alpha_k\left(-\frac{dU_i^\phi}{da_i^\phi}(a_i^\phi[k]) + \max\{0, \lambda_i^\phi[k] \right.\right.$$
$$\left.\left. + \rho g_i^\phi(a_i^\phi[k], \mathbf{r}_{i\cdot}^\phi[k], \mathbf{r}_{\cdot i}^\phi[k])\}\right)\right]_{a_{i,\min}^\phi}^{a_{i,\max}^\phi}, \forall \phi, i \neq \text{dest}(\phi), \quad (21)$$

$$r_{ij}^\phi[k+1] = \left[r_{ij}^\phi[k] - \alpha_k\left(\frac{dV_{ij}^\phi}{dr_{ij}^\phi}(r_{ij}^\phi[k]) \right.\right.$$
$$+ \max\{0, \mu_{ij}[k] + \rho h_{ij}(\mathbf{x}[k], \mathbf{r}_{ij}[k])\}$$
$$- \max\{0, \lambda_i^\phi[k] + \rho g_i^\phi(a_i^\phi[k], \mathbf{r}_{i\cdot}^\phi[k], \mathbf{r}_{\cdot i}^\phi[k])\}$$
$$\left.\left. + \max\{0, \lambda_j^\phi[k] + \rho g_j^\phi(a_j^\phi[k], \mathbf{r}_{j\cdot}^\phi[k], \mathbf{r}_{\cdot j}^\phi[k])\}\right)\right]_{r_{ij,\min}^\phi}^{r_{ij,\max}^\phi},$$
$$\forall \phi, \forall i \neq \text{dest}(\phi), \forall j \neq i. \quad (22)$$

In large robotic networks, distributed algorithms are preferable for their scalability and tolerance to faults [20], [35]. Assume that robot $i \in [n]$ is responsible for updating its position vector $\mathbf{x}_i[k]$, its traffic admission rates $\mathbf{a}_i[k] := \{a_i^\phi[k]\}_\phi$ and outgoing link rates $\mathbf{r}_{i\cdot}[k]$, and the dual variables $\boldsymbol{\lambda}_i[k] := \{\lambda_i^\phi[k]\}_\phi$ and $\boldsymbol{\mu}_{i\cdot}[k] := \{\mu_{ij}[k]\}_j$. Using its own local information, it can immediately update the dual variables $\boldsymbol{\mu}_{i\cdot}$ according to (18). By also keeping track of the incoming traffic rates $\mathbf{r}_{\cdot i}[k]$ from its one-hop neighbors, it can also immediately update the variables $\mathbf{a}_i, \boldsymbol{\lambda}_i$ according to (21) and

(17). Finally, to update its link rates $\mathbf{r}_{ij}$ according to (22) for some neighbor $j \in [n]$, it also needs access to the rates $\mathbf{r}_{j\cdot}[k]$ and $\mathbf{r}_{\cdot j}[k]$ of information sent to and from node j. In other words, all the mentioned variables so far can be updated with at most two-hop information.

The complexity of the robot position update (20) depends on the separability properties of the gradient of the task potential G and on the number of links (l, m) with which robot i might interfere, i.e., for which $\partial c_{lm}/\partial \mathbf{x}_i$ is not zero. For example, if interferences can be neglected, e.g., with a large processing gain S in (6) or an appropriate time and frequency allocation scheme, then the capacity of link (l, m) depends only on the positions of robots l and m, so $\partial c_{lm}/\partial \mathbf{x}_i$ is non-zero only for $l = i$ or $m = i$ and the sum in (20) can be computed from the information collected from the one-hop neighbors. Note that multi-robot task potentials are typically designed to allow distributed gradient computations, see e.g. [3], [21], [22].

B. Choice of Algorithms, Layering and Problem Decomposition

Arguably other optimization algorithms presented in [32] for example could be used to solve (10)-(14) and obtain a trajectory for the robot group, but our choice can be briefly motivated as follows. Barrier methods are popular in robotics [27], [28] and used in [11], [12], but are generally problematic once we introduce uncertainty in the model, in particular the sampling based evaluation of the link capacities in Section IV. Projected gradient algorithms do not work for general capacity functions due to the non-convexity of the constraints (12). Pure penalty methods that do not update the Lagrange multipliers and only increase ρ tend to have poor convergence properties [32]. Multiplier methods are similar to our algorithm but perform a sequence of full primal optimizations and only update the dual variables (and ρ) after a convergence test in the primal problem. Unfortunately, in our case the updates of the configuration variables are the most energy- and time-consuming, since they involve motion, hence should be minimized as much as possible. Moreover, as long as the dual variables are not in the proper region, the primal optimizations can move the robotic network in bad configurations, where the instantaneous communication constraints are far from satisfied and queues build up rapidly. The same issue motivated us not to follow certain two-time scale approaches that update the dual variables at a slower rate, i.e., with $\beta_k = o(\alpha_k)$, see e.g. [36], [37].

It is worth noting that the motion planning algorithm (20) is only coupled to the network layer of the communication system, via the Lagrange multipliers $\boldsymbol{\mu}$ and the link rates $\mathbf{r}$, and not directly to the transport layer adjusting the admission variables $\mathbf{a}$ [30], [38]. The coupling with the network layer is intuitively clear since the choice of configuration essentially controls the capacities of the links. Using an augmented Lagrangian algorithm with $\rho > 0$, instead of dual optimization algorithms more standard in wireless network optimization [30], creates additional couplings between the communication variable updates, but the penalization term is generally required to obtain global convergence results for non-convex problems. Moreover, by increasing it we can approximately

enforce the communication constraints during the transient regime of deployment, see Section V. This in turn leads to smaller delays and shorter queues at the terminals.

C. Convergence

The convergence properties of the primal-dual algorithm can be discussed for the generic form (16)-(18). To simplify the notation, the constraints (11), (12) are denoted $\mathbf{K}(\mathbf{y}) \le 0$ in this section, with $\mathbf{K}(\mathbf{y}) := [K_1(\mathbf{y}), \ldots, K_N(\mathbf{y})]^T$, and corresponding reindexed Lagrange multipliers $\boldsymbol{\xi} = [\xi_1, \ldots, \xi_N]^T$. Let $\mathcal{L}(\mathbf{y}, \boldsymbol{\xi}) := \mathcal{L}_0(\mathbf{y}, \boldsymbol{\xi})$ denote the (non-penalized) Lagrangian function. Finally, recall the definition (19) of the set Ξ of dual variables. Standard convergence results for primal-dual algorithms for non-convex problems are local, and the algorithms must generally be appropriately modified to obtain global convergence guarantees, as briefly described at the end of this section. First, the following properties of a KKT point are generally required in convergence proofs [32]. To simplify the presentation, we only consider local minima that are in the interior of Y, denoted Y°.

Definition 2: A primal point $\mathbf{y}^* \in \mathsf{Y}^\circ$ is called *regular* if the vectors $\nabla K_l(\mathbf{y}^*), l \in [N]$, are linearly independent. Next, define the set of active constraints at some point $\mathbf{y}$ to be $\mathcal{A}(\mathbf{y}) = \{l \in [N] | K_l(\mathbf{y}) = 0\}$. A primal-dual point $(\mathbf{y}^*, \boldsymbol{\xi}^*)$ with $\mathbf{y}^* \in \mathsf{Y}^\circ$ and $\boldsymbol{\xi}^* \in \Xi$ is said to *satisfy the second-order sufficiency conditions* if

$$\nabla_{\mathbf{y}}\mathcal{L}(\mathbf{y}^*, \boldsymbol{\xi}^*)^T = 0; \quad \mathbf{K}(\mathbf{y}^*) \le 0;$$
$$\mathbf{K}(\mathbf{y}^*)^T \boldsymbol{\xi}^* = 0; \quad \xi_l^* > 0, \forall l \in \mathcal{A}(\mathbf{y}^*); \text{ and}$$
$$\mathbf{z}^T \nabla_{\mathbf{yy}}^2 \mathcal{L}(\mathbf{y}^*, \boldsymbol{\xi}^*)\mathbf{z} > 0, \text{ for all } \mathbf{z} \neq 0 \text{ in}$$
$$\{\mathbf{w} | \nabla K_l(\mathbf{y}^*)^T \mathbf{w} = 0, \ \forall l \in \mathcal{A}(\mathbf{y}^*)\}.$$

In the definition above, $\nabla_{\mathbf{yy}}^2 \mathcal{L}$ denotes the matrix of second order derivatives (Hessian) with respect to the primal variables. We then have the following convergence theorem, where all functions in (10)-(14) are assumed twice continuously differentiable.

Theorem 3: Let $\mathbf{y}^*$ be a regular point in Y°, local minimum of (10)-(14), and $\boldsymbol{\xi}^*$ a Lagrange multiplier so that the pair $(\mathbf{y}^*, \boldsymbol{\xi}^*)$ satisfies the second-order sufficiency conditions. Then there exists $\bar{\rho}$ such that for all $\rho \ge \bar{\rho}$, there exists $\bar{\alpha}, \bar{\beta}$ and an open set S containing $(\mathbf{y}^*, \boldsymbol{\xi}^*)$ such that for all $\alpha \in [0, \bar{\alpha}], \beta \in [0, \bar{\beta}]$ and $(\mathbf{y}[0], \boldsymbol{\xi}[0]) \in S$, the sequence $\{(\mathbf{y}[k], \boldsymbol{\xi}[k])\}_k$ following (16)-(18) with $\beta_k = \beta$ and $\alpha_k = \alpha$ for all k remains in S and converges to $(\mathbf{y}^*, \boldsymbol{\xi}^*)$.

Proof: First, with the pair satisfying the second-order sufficiency conditions, there exists $\bar{\rho}$ such that for all $\rho \ge \bar{\rho}$, the matrix $\nabla_{\mathbf{yy}}^2 \mathcal{L}_\rho(\mathbf{y}^*, \boldsymbol{\xi}^*)$ is positive definite [32, p. 304]. The result then follows from [32, Propositions 4.4.2.] ∎

To get theoretical *global* convergence results for the algorithm, we must combine the primal-dual algorithm with other methods mentioned in the previous paragraph, as explained in e.g. [32, Section 4.4.3.]. Multiplier methods in particular are globally convergent methods, which can be used if the primal-dual algorithm fails to converge. In this case, by taking ρ sufficiently large, the communication constraints can be approximately enforced during the primal optimization epochs. A combined multiplier/primal-dual method can be implemented

by varying the stepsizes α_k, β_k and penalty parameter ρ during deployment. We refer the reader to [32, Section 4.4.3.] for a complete discussion of global convergence issues, and to our simulations in Section V illustrating the good practical behavior of the primal-dual algorithm.

IV. Sampling-Based Algorithm

The algorithm presented in Section III requires a model of the average channel capacities $c_{ij}(\mathbf{x})$, but in practice such models are notably difficult to devise and only provide rough approximations [4, chapter 1]. Practical systems must rely on measuring the channel gains (5) and estimating the resulting link capacities. Since UVS can be deployed in a priori unknown environments for which channel strength maps are not available, deployment algorithms should take the channel sampling requirement into account. In this section, we show that this is possible by only modestly increasing the complexity of the primal-dual algorithm.

A. Two-time-scale SPSA Algorithm

We assume for simplicity that we can measure the instantaneous and random channel capacities $c_{ij}[k; \mathbf{x}[k]]$ introduced in Section II-B. In practice, only the SINR (6) or channel gains (5) might be measurable, and the capacities should then be evaluated based on the chosen communication scheme. Recall from Section II-B that $c_{ij}(\mathbf{x})$ is the ergodic limit (8) of the random variables $c_{ij}[k; \mathbf{x}]$, for $\mathbf{x}$ fixed. To implement the iterations (18), (20) and (22) of the algorithm, estimates of the link capacities $c_{ij}(\mathbf{x}[k])$ and of their gradients $\partial c_{lm}(\mathbf{x}[k])/\partial \mathbf{x}_i$ are required, and can only be obtained from the measurements $c_{ij}[k; \mathbf{x}[k]]$. We adopt a two-time scale approach [26, Section 8.6], where the estimates are computed simultaneously with the primal and dual variable updates, albeit with larger stepsizes. Intuitively this device allows us to replace the estimators by their steady-state values in the convergence analysis of the sampling-based algorithm, which then reduces to that of Section III-B.

First consider the following recursive estimators for the link capacities

$$\hat{c}_{lm}[0] = 0, \quad \hat{c}_{lm}[k+1] = \hat{c}_{lm}[k] + \gamma_k(c_{lm}[k; x[k]] - \hat{c}_{lm}[k]), \quad (23)$$

where $\gamma_k \in (0, 1)$ are prespecified stepsizes chosen as discussed in Section IV-B. Next, the capacity gradients are estimated using a form of finite difference approximation. Standard finite-difference methods are not practical in multi-robot systems however, as they are too complex to implement, requiring too much inter-robot coordination. For example, consider for some robot i_1 communicating with some robot i_2 the problem of computing the d-dimensional vector $\partial c_{i_1 i_2}(\mathbf{x}[k])/\partial \mathbf{x}_{i_1}$, which is one of the gradient estimates required to update its position according to (20). Assume for simplicity that the capacities are deterministic and can be perfectly measured, so that $c_{i_1 i_2}(\mathbf{x}[k]) = c_{i_1 i_2}[k; \mathbf{x}[k]]$. With a central-difference scheme, we have for $\delta > 0$ small

$$\forall l \in \{1, \ldots, d\}, \quad \frac{\partial c_{i_1 i_2}(\mathbf{x}[k])}{\partial x_{i_1}^l} \approx$$

$$\frac{c_{i_1 i_2}(\mathbf{x}_{i_1}[k] + \delta \mathbf{e}_l, \mathbf{x}_{-i_1}[k]) - c_{i_1 i_2}(\mathbf{x}_{i_1}[k], \mathbf{x}_{-i_1}[k])}{\delta}, \quad (24)$$

where $\mathbf{e}_l$ is the lth unit vector, and $\mathbf{x}_{-i_1}$ is the $(n-1)d$ dimensional vector of coordinates of the robots $j \neq i_1$. Constructing approximation (24) requires that robot i_1 move d times around $\mathbf{x}_{i_1}[k]$, to the positions $\mathbf{x}_{i_1}[k] + \delta \mathbf{e}_l$, $l = 1, \ldots, d$, while the other robots remain immobile. All robots must execute these small individual motions, one after the other, resulting in unreasonably long update times to compute just one iteration of (20).

This issue can be resolved using a stochastic sampling strategy known as Simultaneous Perturbation Stochastic Approximation (SPSA), see [25], [39], [40] and the references therein. We divide period k into two subperiods. In the first subperiod, the robots are in the configuration $\mathbf{x}[k]$, and each robot measures the link capacities $c_{lm}[k; \mathbf{x}[k]]$ that its presence influences. It also generates for that period, independently of the other robots, a random d-dimensional vector, say $\boldsymbol{\Delta}_i[k]$ for robot i, with independent and identically distributed (iid) entries in $\{+1, -1\}$ such that $P(\Delta_i^l = 1) = P(\Delta_i^l = -1) = 1/2$, $l = 1, \ldots, d$. Let $\boldsymbol{\Delta}[k] = [\boldsymbol{\Delta}_1[k]^T, \ldots, \boldsymbol{\Delta}_n[k]^T]^T$ denote the aggregate nd dimensional random vector. In the second subperiod, denoted k^+, all robots move simultaneously and randomly, with robot i moving by an amount $\delta \boldsymbol{\Delta}_i[k]$ from $\mathbf{x}_i[k]$, where δ is a small constant. Then, each robot again measures the relevant link capacities $c_{lm}[k^+; \mathbf{x}[k] + \delta \boldsymbol{\Delta}[k]]$. They can now update the following sequences

$$\hat{\mathbf{d}}_{lm,i}[k+1] = \hat{\mathbf{d}}_{lm,i}[k] + \gamma_k\big(\boldsymbol{\Delta}_i[k](c_{lm}[k^+; \mathbf{x}[k] + \delta \boldsymbol{\Delta}[k]] - c_{lm}[k; \mathbf{x}[k]]) - \hat{\mathbf{d}}_{lm,i}[k]\big), \quad (25)$$

where the stepsizes γ_k are the same as in (23), and $\hat{\mathbf{d}}_{lm,i}[0] = \mathbf{0}$.

The final sampling-based algorithm followed by the robots is the same as the primal-dual algorithm (20)-(22), (17)-(18), except that in (20), (18) and (22) the capacities $c_{ij}(\mathbf{x}[k])$ in the expression $h_{ij}(\mathbf{x}[k], \mathbf{r}_{ij}[k])$ are replaced by the estimates $\hat{c}_{ij}[k]$ from (23) and in (20) the terms $\partial c_{lm}(\mathbf{x}[k])/\partial \mathbf{x}_i$ are replaced by the estimates $\hat{\mathbf{d}}_{lm,i}[k]/\delta$ based on (25). Hence SPSA drastically accelerates the gradient estimation procedure by allowing all robots to sample the channels simultaneously. Moreover extensive numerical experiments [41] have shown that for many problems the total number of iterations to reach convergence with this 2-sample randomized approximation scheme is often of the same order as the one required with the standard central difference scheme.

To understand the motivation behind (25), according to Theorem 5 below, for appropriate small stepsizes γ_k and α_k, the estimates $\hat{\mathbf{d}}_{lm,i}[k]/\delta$ used in (20) in place of $\partial c_{lm}(\mathbf{x}[k])/\partial \mathbf{x}_i$, approximately contribute the following term to the dynamics of $\mathbf{x}[k]$ (see (28) below)

$$\mathbb{E}\left[\boldsymbol{\Delta}_i[k]\frac{c_{lm}(\mathbf{x}[k] + \delta \boldsymbol{\Delta}[k]) - c_{lm}(\mathbf{x}[k])}{\delta}\Big| \mathbf{x}[k]\right], \quad (26)$$

which by a Taylor expansion of $c_{lm}(\mathbf{x})$ is equal to

$$\mathbb{E}\Big[\boldsymbol{\Delta}_i[k]\boldsymbol{\Delta}[k]^T \nabla c_{lm}(\mathbf{x}[k]) +$$
$$\frac{1}{2}\delta \boldsymbol{\Delta}_i[k]\boldsymbol{\Delta}[k]^T \nabla^2 c_{lm}(\mathbf{x}[k] + s\delta \boldsymbol{\Delta}[k])\boldsymbol{\Delta}[k]\Big], \quad s \in [0, 1]$$
$$= \partial c_{lm}(\mathbf{x}[k])/\partial \mathbf{x}_i + O(\sup_{\mathbf{z}} \delta \|\nabla^2 c_{lm}(\mathbf{z})\|), \quad (27)$$

since $\mathbb{E}\big[\boldsymbol{\Delta}_i[k]\boldsymbol{\Delta}[k]^T\big] = [0 \dots I_d \dots 0]$.

Remark 4: Various choices for the random directions $\boldsymbol{\Delta}_i[k]$ are valid, see [25], [26, Section 10.7], and [42] for experimental considerations on the selection of random directions for robotic applications. Moreover with δ smaller than the vehicle size (see next paragraph), no explicit motion at subperiod k^+ is necessary, and we can rely instead on several antennas mounted at the right positions on the vehicle. For the scheme above, 5 antennas at coordinates $(0,0), (\pm\delta, \pm\delta)$ in the local vehicle frame could be used. At subperiod k, the antenna at $(0,0)$ measures the channels, and at subperiod k^+, one of the other antennas is chosen randomly to obtain additional channel measurements necessary for the gradient estimation procedure.

B. Convergence of the Sampling-Based Algorithm

We now discuss a convergence result for the primal-dual algorithm (20)-(22), (17)-(18) using the estimates (23), (25). Related multiple time-scale algorithms for simulation-based optimization with decreasing stepsizes are discussed in e.g. [37], [43], but here we focus our analysis on the more practical case of constant stepsizes, see also [44]. Hence for small constants $\epsilon, \gamma, \delta \in (0,1)$, assume that for all k we have $\alpha_k = \beta_k = \epsilon$, $\gamma_k = \gamma$, in (20)-(22), (17)-(18), (23), (25). Define the piecewise-constant process $\boldsymbol{\theta}^{\epsilon,\gamma}(\cdot) := (\mathbf{y}^{\epsilon,\gamma}(\cdot), \boldsymbol{\xi}^{\epsilon,\gamma}(\cdot))$ tracking the primal-dual iterates

$$\mathbf{y}^{\epsilon,\gamma}(t) = \mathbf{y}[k], \; \boldsymbol{\xi}^{\epsilon,\gamma}(t) = \boldsymbol{\xi}[k], \;\; \forall t \in [k\epsilon, (k+1)\epsilon), \forall k.$$

The following convergence result applies as the stepsizes ϵ, γ tend to zero with $\epsilon \ll \gamma$. For $a < b$ in $\mathbb{R}$, we use the notation $\dot{x} = [f(x)]_a^b$ to mean the ordinary differential equation (ode)

$$\dot{x} = \begin{cases} f(x), & \text{if } a < x < b, \text{or } x = a \text{ and } f(x) \geq 0, \\ & \quad \text{or } x = b \text{ and } f(x) \leq 0 \\ 0, & \text{if } x = a \text{ and } f(x) < 0, \text{or } x = b \text{ and } f(x) > 0. \end{cases}$$

The reflection terms $\mathbf{z}_i(\cdot)$ in (28) below serve the same purpose of keeping the trajectories of an ode in the set W in higher dimensions, see [26] for a detailed definition.

Theorem 5: Fix $\delta \in (0,1)$. Assume that the sequences $\{\hat{c}_{lm}[k]\}_k$, $\{\hat{\mathbf{d}}_{lm,i}[k]\}_k$, $\{\lambda_i^\phi[k]\}_k$, $\{\mu_{lm}[k]\}_k$ are bounded almost surely, for all $l, m, i \in [n]$, $\phi \in \Phi$. Moreover, assume that the measured capacities $\{c_{ij}[k; \mathbf{x}] | k \geq 0, \mathbf{x} \in \mathsf{W}\}$ are uniformly integrable, i.e.,

$$\lim_{M \to \infty} \sup_{k, \mathbf{x}} \mathbb{E}(|c_{ij}[k; \mathbf{x}]| \mathbf{1}_{\{|c_{ij}[k;\mathbf{x}]| > M\}}) = 0.$$

Then for any $T > 0, \eta > 0$, and any sequence $\{\boldsymbol{\theta}^{\epsilon,\gamma}(\cdot)\}_{\epsilon,\gamma}$ with $\gamma \to 0$, $\epsilon/\gamma \to 0$, there is a deterministic process $t \to \bar{\boldsymbol{\theta}}^\delta(t) := (\mathbf{x}(t), \mathbf{a}(t), \mathbf{r}(t), \boldsymbol{\lambda}(t), \boldsymbol{\mu}(t))$ such that we have

$$\lim_{\gamma \to 0, \, \epsilon/\gamma \to 0} \mathbb{P}\left(\sup_{0 \leq t \leq T} \|\boldsymbol{\theta}^{\epsilon,\gamma}(t) - \bar{\boldsymbol{\theta}}^\delta(t)\| > \eta \right) = 0,$$

where $t \to \bar{\boldsymbol{\theta}}^\delta(t)$ satisfies the differential equations

$$\dot{\mathbf{x}}_i = -\frac{\partial G}{\partial \mathbf{x}_i} + \sum_{l, m \neq l} \max\{0, \mu_{lm} + \rho h_{lm}(\mathbf{x}, \mathbf{r}_{lm})\} \times$$
$$\left(\frac{\partial c_{lm}}{\partial \mathbf{x}_i}(\mathbf{x}) + \mathbf{b}_{lm,i} \right) + \mathbf{z}_i, \tag{28}$$

$$\dot{a}_i^\phi = \left[\frac{dU_i^\phi(a_i^\phi)}{da_i^\phi} - \max\{0, \lambda_i^\phi + \rho g_i^\phi(a_i^\phi, \mathbf{r}_{i\cdot}^\phi, \mathbf{r}_{\cdot i}^\phi)\} \right]_{a_{i,\min}^\phi}^{a_{i,\max}^\phi},$$

$$\dot{r}_{ij}^\phi = \Big[-\frac{dV_{ij}^\phi(r_{ij}^\phi)}{dr_{ij}^\phi} - \max\{0, \mu_{ij} + \rho h_{ij}(\mathbf{x}, \mathbf{r}_{ij})\}$$
$$+ \max\{0, \lambda_i^\phi + \rho g_i^\phi(a_i^\phi, \mathbf{r}_{i\cdot}^\phi, \mathbf{r}_{\cdot i}^\phi)\}$$
$$- \max\{0, \lambda_j^\phi + \rho g_j^\phi(a_j^\phi, \mathbf{r}_{j\cdot}^\phi, \mathbf{r}_{\cdot j}^\phi)\} \Big]_{r_{ij,\min}^\phi}^{r_{ij,\max}^\phi},$$

$$\dot{\lambda}_i^\phi = \left[g_i^\phi(a_i^\phi, \mathbf{r}_{i\cdot}^\phi, \mathbf{r}_{\cdot i}^\phi) \right]_0^{\lambda_{i,\max}^\phi},$$
$$\dot{\mu}_{ij} = \left[h_{ij}(\mathbf{x}, \mathbf{r}_{ij}) \right]_0^{\mu_{ij,\max}}.$$

Here $\mathbf{z}_i(\cdot)$ are reflection terms keeping $\mathbf{x}_i(\cdot)$ in W (equal to zero if $\mathbf{x}_i$ is in the interior of W), and $\mathbf{b}_{lm,i}(\cdot)$ are bias terms due to the finite difference approximations (26), satisfying $\sup_t \|\mathbf{b}_{lm,i}(t)\| = O(\delta \sup_{\mathbf{x} \in \mathsf{W}} \|\nabla^2 c_{lm}(\mathbf{x})\|)$.

Proof (sketch): For any given configuration vectors $\mathbf{x} \in \mathsf{W}^n$, and $\boldsymbol{\Delta}$ an nd-dimensional random vector generated as in the SPSA algorithm, the differential equations

$$\dot{\zeta}_{lm}(t) = c_{lm}(\mathbf{x}) - \zeta_{lm}(t),$$
$$\dot{\boldsymbol{\eta}}_{lm,i}(t) = \mathbb{E}[\boldsymbol{\Delta}_i(c_{lm}(\mathbf{x} + \delta\boldsymbol{\Delta}) - c_{lm}(\mathbf{x}))] - \boldsymbol{\eta}_{lm,i}(t),$$

corresponding to the iterations (23), (25) respectively, each have a unique globally asymptotically stable equilibrium, namely $\zeta_{lm}(t) \equiv c_{lm}(\mathbf{x})$ and $\boldsymbol{\eta}_{lm,i}(t) \equiv \mathbb{E}[\boldsymbol{\Delta}_i(c_{lm}(\mathbf{x}+\delta\boldsymbol{\Delta}) - c_{lm}(\mathbf{x}))]$ respectively. The theorem is then a consequence of [26, Theorem 6.2]. Intuitively, we can treat the estimates as having converged in the limit ode due to the choice of relative stepsizes. The approximation order of the bias terms in the gradient estimates is obtained from the computation (27). The other technical assumptions required in [26, Theorem 6.2] are consequences of our boundedness and uniform integrability assumptions. ∎

Except for the bias term in (28), the ode in Theorem 5 is the continuous-time equivalent of the primal-dual algorithm, whose local convergence to a KKT point is discussed in Section III-C. The issues with global convergence in the absence of convexity assumptions are the same as discussed at the end of that section, and in particular to improve robustness we might have to increase the penalty parameter ρ, or update the Lagrange multipliers more slowly, as in [37].

Decreasing δ leads to smaller bias in the gradient estimates, however their variance increases since a small δ in (26) amplifies the fluctuations in the instantaneous capacity measurements. A practical choice of δ depends on the level of noise in these measurements. Note also that the noisy capacity and gradient estimates are multiplied by the penalty parameter ρ. Hence as we increase ρ to increase the region of convergence, the bias and noise are amplified. For this reason, once close to an equilibrium configuration (e.g., no visible
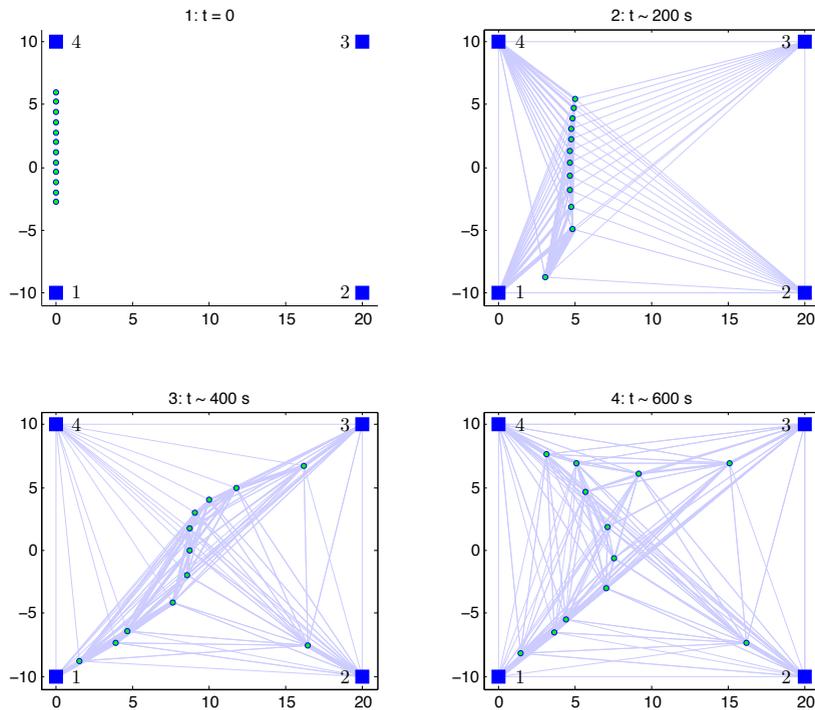
Fig. 1.   Scenario 1: using mobility to deploy an ad-hoc communication network. BS 2, 3, 4 (the large squares) must send information at unit rate to BS 1 but are too far away to achieve this rate. Hence there is a single flow to support, with all information directed toward BS 1. The link capacities follow the model (29). 12 mobile robots (the small discs) are deployed using the primal-dual algorithm, with primal and dual variables updated every 50 ms in real time. If one unit of length on the figure represents 100 m, the maximum velocity of the vehicles is 50 m/s (180 km/hr). A group configuration achieving the desired communication rates is reached after about 8 minutes in real deployment time (about 9600 iterations of the algorithm), with $\rho = 5, \alpha = 0.02, \beta = 0.2$. The relative magnitudes of the link rates are represented by the thickness of the lines between terminals.
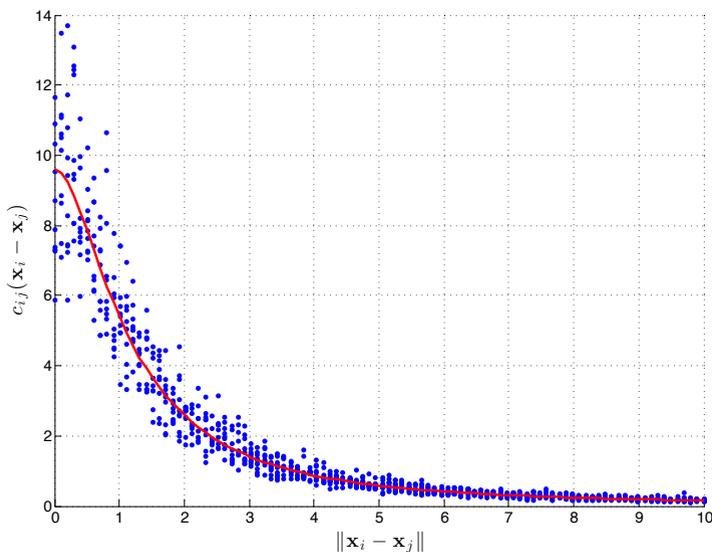
Fig. 2.   Link capacity function (29) used in the simulations (solid curve), with random samples for Scenario 2.

Fig. 3.   Trajectories of the routing constraint values $g_i^\phi(a_i^\phi[k], \mathbf{r}_{i\cdot}^\phi[k], \mathbf{r}_{\cdot i}^\phi[k])$ for Scenario 1. In the initial configuration the desired communication rates are not feasible, but the communication constraints are satisfied about 8 min after the network starts deploying.

progress is made), one should decrease again ρ, possibly setting $\rho = 0$ to revert back to a pure primal-dual method.

V. SIMULATIONS

We now illustrate the practical behavior of the communication-constrained deployment algorithm in three scenarios. All simulations are performed on a single computer executing the primal-dual updates for all vehicles.

First, Fig. 1 shows snapshots of a robot deployment trajectory for a problem with no task potential (i.e., $G(\mathbf{x}) = 0$), where the robots are only used to establish an ad-hoc communication network. They relay the communications between fixed Base Stations (BS) that are too far from each other to support the requested rates. The capacity functions are assumed known and chosen as

$$c_{ij}(\mathbf{x}_i, \mathbf{x}_j) = 4 \ln(1 + p(\mathbf{x}_i, \mathbf{x}_j)), \qquad (29)$$

$$\text{with } p(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{0.1 + 0.25\|\mathbf{x}_i - \mathbf{x}_j\|^2}, \qquad (30)$$
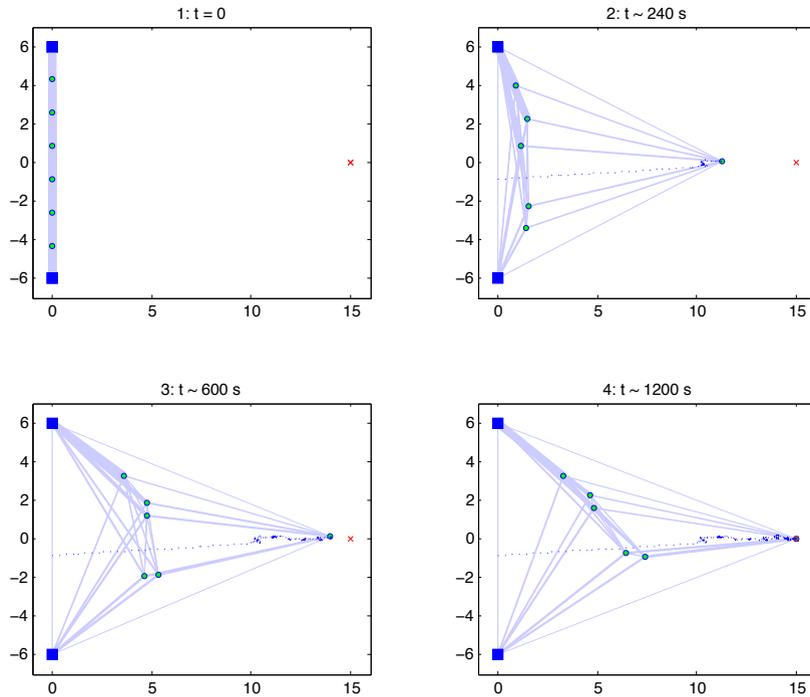
Fig. 4. In Scenario 2, one designated robot must approach the target denoted by a cross, placed at $x = 1.5$ km (the dotted curve shows the robot trajectory). It generates information for the top BS at unit rate. The bottom BS also transmits information to the top BS at unit rate, again as part of the same and unique flow. The other robots spread to establish the necessary links. The primal and dual variables are updated every 50 ms, using the measurement-based algorithm and no knowledge of the link capacity model. The maximum velocity of the vehicles is 10 m/s (36 km/hr), as could be appropriate for ground robots.
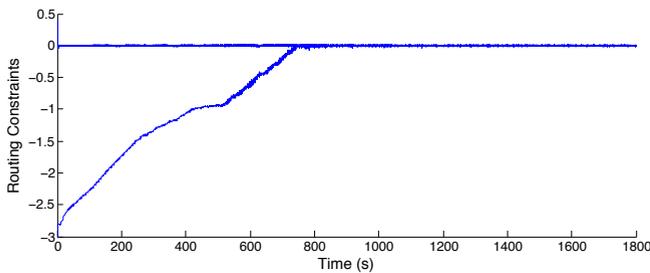
Fig. 5. Trajectories of the routing constraint values $g_i^\phi(a_i^\phi[k], \mathbf{r}_i^\phi[k], \mathbf{r}_{\cdot i}^\phi[k])$ for Scenario 2. Note that here the communication constraints are essentially always satisfied during deployment, i.e., the constraint values remain non-positive.

for all links (i,j), see Fig. 2 and Section II-B. In particular, it is assumed here that the communication scheme establishes interference-free links. Note that the quadratic power decay in (30) is a frequently assumed model, corresponding to channels with direct line-of-sight propagation only [4, chap. 2].

Fig. 3 shows the evolution of the instantaneous values of the functions $g_i^\phi(a_i^\phi[k], \mathbf{r}_i^\phi[k], \mathbf{r}_{\cdot i}^\phi[k])$, whose average values must be non-positive at steady-state (see (7)). A spatial configuration for the robots satisfying the desired flow rates is found after about 8 minutes of real time deployment, assuming the physical parameters given on Fig. 1. Note that since the link capacity model is known, the algorithm can alternatively be run offline for a few seconds (about 10000 iterations) to obtain open-loop trajectories or simply feasible link rates and final positions for the robots to reach.

In Scenario 2 depicted on Fig. 4, one robot must approach a given waypoint at a known position q^*, which is achived with the task potential (2), while transmitting back some information relayed to the base station by the other robots. The robots must also establish a communication flow between this base station and another distant one, as in the previous scenario. The instantaneous channel capacities are now stochastic and simulated as $c_{ij}[k; \mathbf{x}] = \eta_{ij}[k] c_{ij}(\mathbf{x}_i, \mathbf{x}_j)$, with $c_{ij}(\mathbf{x}_i, \mathbf{x}_j)$ given by (29) and $\{\eta_{ij}[k]\}_k$ an iid sequence of log-normal random variables with distribution $\ln \mathcal{N}(0, 0.2)$, see Fig. 2. Here the robots do not know the model (29) and can only rely on the fluctuating measurements $c_{ij}[k; \mathbf{x}]$ taken during deployment to update their positions and the communication rates. They implement the two-time-scale SPSA algorithm described in Section IV-A, and Fig. 6 illustrates the trajectory of the capacity estimates (23) for one link. As can be seen from Fig. 5, the instantaneous routing constraints corresponding to (11) (and in fact also the instantaneous capacity constraints corresponding to (12)) are essentially satisfied during the whole deployment, resulting in short communication delays and small queues at the terminals. With a variance here of about 0.27 for the η_{ij} variables, the algorithm exhibits a satisfying converging behavior. More generally the performance of such a finite difference scheme can be sensitive to the level of measured noise, and additional averaging over several capacity measurements before moving might be necessary. Variance reduction in the estimates (23), (25) can also be achieved by reducing the values of α, β, γ, similarly associated with slower motion and slower convergence rate of these estimates. The important role of the choice of stepsizes on the convergence
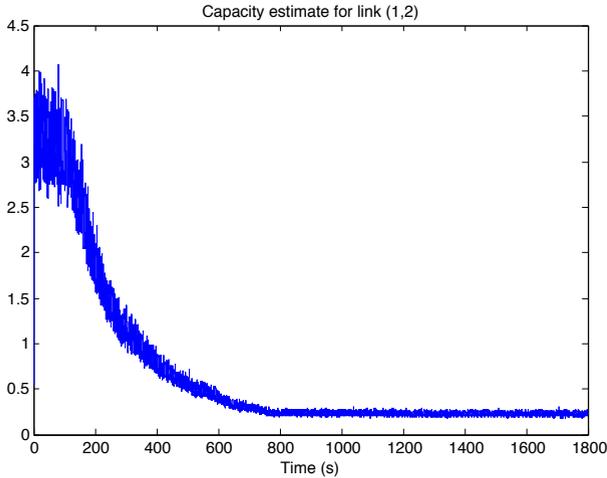
Fig. 6.   Sample trajectory of the capacity estimates (23) for the link $(1, 2)$ in Scenario 2, with $\gamma = 0.2$.

Fig. 7.   Duration of deployment vs. choice of stepsize $\alpha_k \equiv \alpha$ for Scenario 2. We take $\beta_k \equiv \beta = 10\alpha$, $\gamma_k \equiv 0.2$ and $\delta_k \equiv 0.01$, $\rho = 5$ in all simulations. The number of iterations of the primal-dual algorithm is converted to deployment time by assuming that variables are updated every 50 ms. The duration indicated is the time until an apparent steady state configuration is reached by the algorithm. For $\alpha > 0.05$, the algorithm becomes unstable, i.e., does not converge to a locally optimum configuration.

time of the algorithm, or equivalently, on the time necessary to deploy the robotic network, is briefly illustrated on Fig. 7.

Finally, Scenario 3 is a communication-constrained coverage control problem, i.e., the task potential is (3), with the probability distribution of the event location Z assumed uniform over a square region W, and with the cost $c(x) = x^2$. The distributed computation of the gradient of the task potential is described in [22]. Twelve robots are used for coverage and required to maintain communication with a base station at a fixed rate. In this scenario we simulate the effect of interferences between vehicles by taking the capacity functions to be

$$c_{ij}(\mathbf{x}) = 4 \ln \left(1 + \frac{p(\mathbf{x}_i, \mathbf{x}_j)}{0.05 + G \sum_{l \neq i} p(\mathbf{x}_l, \mathbf{x}_j)} \right), \qquad (31)$$

with the received powers $p(\mathbf{x}_l, \mathbf{x}_m)$ given by (30), and $G = 0.05$ (low interferences) or $G = 0.2$ (high interferences). Note that it is assumed in (31) that the links originating from a given transmitter do not interfere with each other, for example through the use of different frequency tones or time slots. The instantaneous capacities are simulated as in Scenario 2 by multiplying the capacity functions with log-normal random variables, and the algorithm used is again the measurement-based SPSA version. Fig. 8 shows the result of deployment for the high and low interference regimes. As interferences increase, the robots need to establish more shorter links and rely on multi-hop communications to satisfy the rate requirements, and the robots closer to the base station relay more traffic. Note also that transmission power control could be included in our framework to help manage interferences [4, chap. 16].

VI. CONCLUSIONS

A communication constrained robot deployment algorithm has been presented to jointly optimize the steady-state locations of a group of unmanned vehicles and the wireless communication network necessary to support their mission. An optimization-based approach is proposed, where a general augmented Lagrangian primal-dual algorithm can drive the robotic network to a locally optimal final configuration. A important aspect of this approach is that it can be naturally used as a measurement based feedback strategy, when no sufficiently reliable prior model of the communication channels is available. Future work includes adapting the algorithm to better accommodate vehicles with significant dynamics, and optimizing the choice of algorithm parameters in different scenarios based on field experiments.

REFERENCES

[1] "Unmanned aircraft systems roadmap 2010-2035," U. S. Army UAS Center of Excellence, Tech. Rep., 2010.

[2] P. Ögren, E. Fiorelli, and N. Leonard, "Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment," *IEEE Trans. Autom. Contr.*, vol. 49, no. 8, pp. 1292–1302, August 2004.

[3] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control," *IEEE Control Syst. Mag.*, vol. 27, no. 2, pp. 71–82, April 2007.

[4] A. Goldsmith, *Wireless Communications*. Cambridge University Press, 2005.

[5] J. Cortés, S. Martínez, and F. Bullo, "Spatially-distributed coverage optimization and control with limited-range interactions," *ESAIM: Control, Optimisation and Calculus of Variations*, vol. 11, no. 4, pp. 691–719, 2005.

[6] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.

[7] Y. Mostofi, A. Gonzalez-Ruiz, A. Ghaffarkhah, and D. Li, "Characterization and modeling of wireless channels for networked robotic and control systems - a comprehensive overview," in *Proc. International Conference on Intelligent Robots and Systems (IROS)*, St. Louis, MO, 2009, pp. 4849–4854.

[8] A. Ghaffarkhah and Y. Mostofi, "Communication-aware navigation functions for cooperative target tracking," in *Proc. American Control Conference*, ST. Louis, MO, June 2009, pp. 1316–1322.

[9] M. Lindhé and K. H. Johansson, "Adaptive exploitation of multipath fading for mobile sensors," in *IEEE International Conference on Robotics and Automation*, Anchorage, AK, May 2010, pp. 1934–1939.

[10] S. Firouzabadi and N. C. Martins, "Jointly optimal power allocation and constrained node placement in wireless networks of agents," University of Maryland, Tech. Rep., 2008.

[11] M. M. Zavlanos, A. Ribeiro, and G. J. Pappas, "Mobility and routing control in networks of robots," in *Proc. 49th IEEE Conference on Decision and Control*, Atlanta, GA, December 2009, pp. 7545–7550.

(a) High Interferences ($G = 0.2$)

(b) Low Interferences ($G = 0.05$)
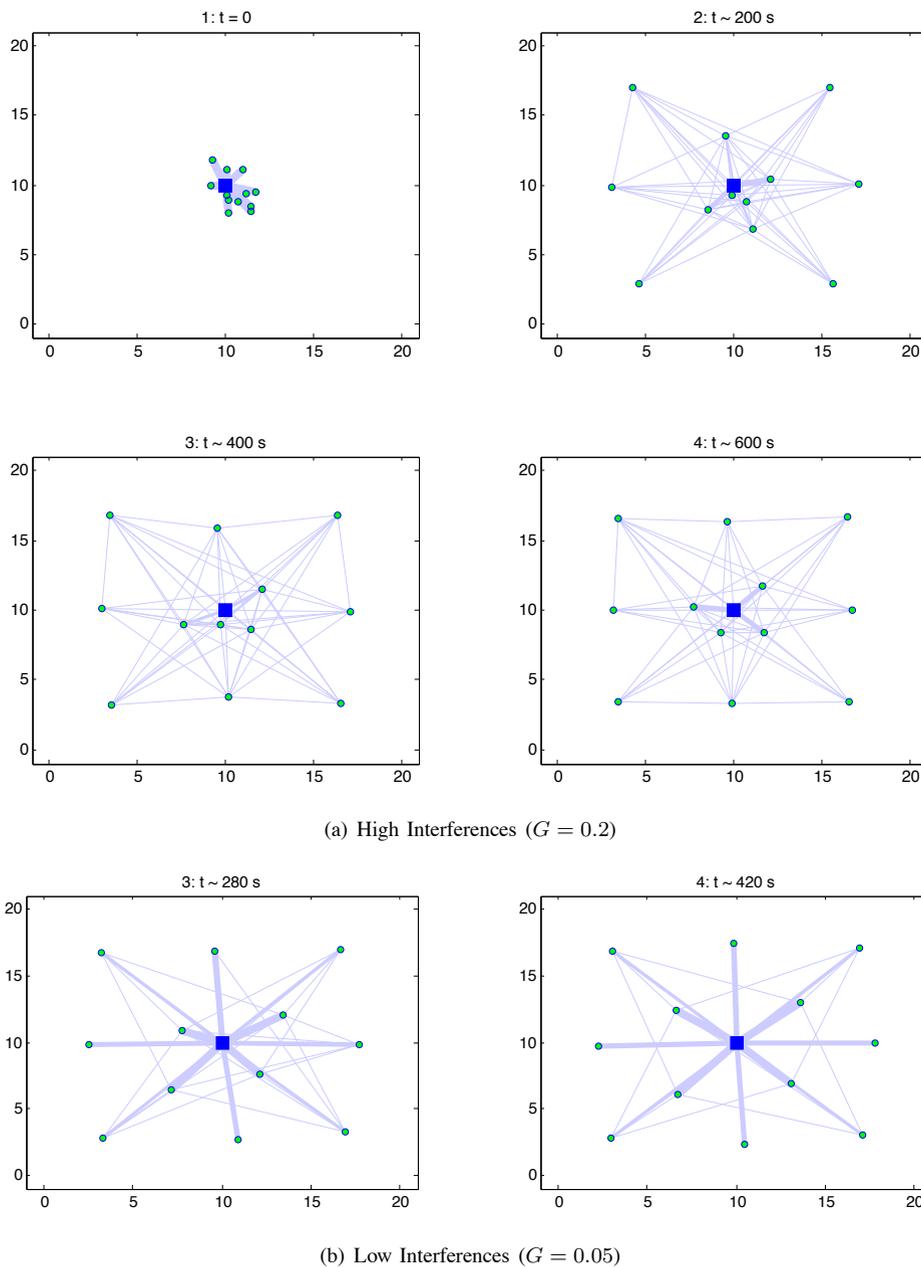
Fig. 8. In Scenario 3, a group of 12 robots starts around a BS and executes a coverage control deployment (task potential (3)), for an event location distribution uniform in the square $[0, 20] \times [0, 20]$. All robots must maintain communication with the base station and transmit back information at a rate of 0.4. As the level of interferences increases, this requires that the robots send their packets through more routes, and that some robots remain closer to the BS to act as relays with higher capacity links.

[12] ——, "Distributed control of mobility and routing in networks of robots," in *Proceedings of the IEEE Workshop on Signal Processing: Advances in Wireless Communications*, Marrakech, Morocco, June 2010.

[13] J. Fink, A. Ribeiro, V. Kumar, and B. Sadler, "Optimal robust multihop routing for wireless networks of mobile micro autonomous systems," in *Military Communications Conference*, San Jose, CA, November 2010, pp. 1268–1273.

[14] E. M. Craparo, J. P. How, and E. Modiano, "Throughput optimization in mobile backbone networks," *IEEE Trans. Mobile Comput.*, vol. 10, no. 4, pp. 560–572, April 2011.

[15] M. Hsieh, A. Cowley, V. Kumar, and C. Taylor, "Maintaining network connectivity and performance in robot teams," *J. Field Robotics*, vol. 25, no. 1-2, pp. 111–131, January 2008.

[16] H. C. H. Chiu, B. Ryu, H. Zhu, P. Szekely, R. Maheswaran, C. Rogers, A. Galstyan, B. Salemi, M. Rubenstein, and W.-M. Shen, "Tentacles: self-configuring robotic radio networks in unknown environments," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Piscataway, NJ, 2009, pp. 1383–1388.

[17] M. A. Vieira, R. Govindan, and G. S. Sukhatme, "Towards autonomous wireless backbone deployment in highly-obstructed environments," in *IEEE International Conference on Robotics and Automation*, Shangai, China, May 2011.

[18] J. Fink, N. Michael, A. Kushleyev, and V. Kumar, "Experimental characterization of radio signal propagation in indoor engironments with application to estimation and control," *Proc. International Conference on Intelligent Robots and Systems (IROS 2009)*, pp. 2834–2839, October 2009.

[19] J. Fink, A. Ribeiro, and V. Kumar, "Robust control for mobility and wireless communication in cyber-physical systems with application to robot teams," *Proc. IEEE*, vol. 100, no. 1, pp. 150–163, January 2012.

[20] F. Bullo, J. Cortés, and S. Martinez, *Distributed Control of Robotic Networks*. Princeton University Press, 2009.

[21] J. Reif and H. Wang, "Social potential fields: A distributed behavioral control for autonomous robots," *Robot. Autonomous Systems*, vol. 27, pp. 171–194, 1999.

[22] J. Cortés, S. Martínez, T. Karatas, and F. Bullo, "Coverage control for

mobile sensing networks," *IEEE Trans. Robot. Autom.*, vol. 20, no. 2, pp. 243–255, April 2004.

[23] E. Frazzoli and F. Bullo, "Decentralized algorithms for vehicle routing in a stochastic time-varying environment," in *Proc. Conference on Decision and Control*, 2004.

[24] L. C. A. Pimenta, M. Schwager, Q. Lindsey, V. Kumar, D. Rus, R. C. Mesquita, and G. A. S. Pereira, "Simultaneous coverage and tracking (scat) of moving targets with robot networks," in *Proc. Eighth International Workshop on the Algorithmic Foundations of Robotics*, 2008.

[25] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation," *IEEE Trans. Autom. Contr.*, vol. 37, no. 3, pp. 332–341, March 1992.

[26] H. J. Kushner and G. G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*, 2nd ed. Springer, 2003.

[27] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *International Journal of Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986.

[28] E. Rimon and D. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Trans. Robot. Autom.*, vol. 8, no. 5, pp. 501–518, October 1992.

[29] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2325–2383, October 1998.

[30] A. Ribeiro and G. Giannakis, "Separation principles in wireless networking," *IEEE Trans. Inf. Theory*, vol. 56, no. 9, pp. 4488–4505, September 2010.

[31] A. Ribeiro, "Stochastic learning algorithms for optimal design of wireless fading networks," in *Proc. IEEE Workshop on Signal Processing and Advances in Wireless Communication*, Marrakech, Morocco, June 2010.

[32] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Athena Scientific, 1999.

[33] H. J. Kushner and D. S. Clark, *Stochastic Approximation Methods for Constrained and Unconstrained Systems*. Springer-Verlag, 1978.

[34] A. Ribeiro, "Ergodic stochastic optimization algorithms for wireless communication and networking," *IEEE Trans. Signal Processing*, vol. 58, no. 12, pp. 6369–6386, December 2010.

[35] L. Parker, "Heterogeneous multi-robot cooperation," Ph.D. dissertation, Massachusetts Institute of Technology, 1994.

[36] V. S. Borkar, "An actor-critic algorithm for constrained Markov decision processes," *Systems and Control Letters*, vol. 54, pp. 207–213, 2005.

[37] S. Bhatnagar, N. Hemachandra, and V. K. Mishra, "Stochastic approximation algorithms for constrained optimization via simulation," *ACM Trans. Modeling and Computer Simulation*, vol. 21, no. 3, 2011.

[38] M. Chiang, S. H. Low, R. A. Calderbank, and J. C. Doyle, "Layering as optimization decomposition," *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, January 2007.

[39] H. F. Chen, T. E. Duncan, and B. Pasik-Duncan, "A Kiefer-Wolfowitz algorithm with randomized differences," *IEEE Trans. Autom. Contr.*, vol. 44, no. 3, pp. 442–453, March 1999.

[40] J. C. Spall, "Adaptive stochastic approximation by the simultaneous pertubation method," *IEEE Trans. Autom. Contr.*, vol. 45, no. 10, pp. 1839–1853, October 2000.

[41] ——, *Introduction to Stochastic Search and Optimization*. Wiley, 2003.

[42] N. Atanasov, J. Le Ny, N. Michael, and G. J. Pappas, "Stochastic source seeking in complex environments," in *Proc. IEEE Conference on Robotics and Automation*, St. Paul, MN, May 2012.

[43] S. Bhatnagar, M. C. Fu, S. I. Marcus, and S. Bhatnagar, "Two-timescale algorithms for simulation optimization of hidden Markov models," *IEE Trans.*, vol. 33, no. 3, pp. 245–258, March 2001.

[44] F. J. Vázquez-Abad and V. Krishnamurthy, "Self learning control of constrained Markov decision processes - a gradient approach," Les cahiers du GERAD, Tech. Rep., 2003.

Jerome Le Ny (S'05-M'09) is an assistant professor in the Department of Electrical Engineering at the École Polytechnique de Montréal since May 2012. He received the B.S. degree from the Ecole Polytechnique, France, in 2001, the M.Sc. degree in Electrical Engineering from the University of Michigan, Ann Arbor, in 2003, and the Ph.D. degree in Aeronautics and Astronautics from the Massachusetts Institute of Technology, Cambridge, in 2008. From 2008 to 2012 he was a Postdoctoral Researcher with the GRASP Laboratory at the University of Pennsylvania. His research interests include robust and stochastic control with applications to autonomous and embedded systems, air traffic control, and the associated computational methods supporting the design of complex systems.

Alejandro Ribeiro (M'07) is an assistant professor at the Department of Electrical and Systems Engineering at the University of Pennsylvania (Penn), Philadelphia, where he started in 2008. He received the B.Sc. in electrical engineering from the Universidad de la Republica Oriental del Uruguay, Montevideo, in 1998. From 2003 to 2008 he was at the Department of Electrical and Computer Engineering, the University of Minnesota, Minneapolis, where he received the M.Sc. and Ph.D. in electrical engineering. From 1998 to 2003 he was a member of the technical staff at Bellsouth Montevideo. His research interests lie in the areas of communication, signal processing, and networking. His current research focuses on network and wireless communication theory. He received the 2012 S. Reid Warren, Jr. Award presented by Penn's undergraduate student body for outstanding teaching and the NSF CAREER award in 2010. He is also a Fulbright scholar and the recipient of student paper awards at ICASSP 2005 and ICASSP 2006.

George J. Pappas (S'90-M'91-SM'04-F'09) received the Ph.D. degree in electrical engineering and computer sciences from the University of California, Berkeley in 1998, for which he received the Eliahu Jury Award for Excellence in Systems Research. He is currently the Joseph Moore Professor of Electrical and Systems Engineering at the University of Pennsylvania, Philadelphia. He is a member of the General Robotics, Automation, Sensing and Perception (GRASP) Laboratory and the PRECISE Center for Embedded Systems. His current research interests include hybrid and embedded systems, hierarchical control systems, distributed control systems, nonlinear control systems, with applications to robotics, unmanned aerial vehicles, biomolecular networks, and green buildings. Dr. Pappas has received numerous awards, including the National Science Foundation (NSF) CAREER Award in 2002, the NSF Presidential Early Career Award for Scientists and Engineers in 2002, the 2009 George S. Axelby Outstanding Paper Award, and the 2010 Anotnio Ruberti Outstanding Young Researcher Prize.