INVITED PAPER

Robust Control for Mobility and Wireless Communication in Cyber–Physical Systems With Application to Robot Teams

Control of an autonomous robot team with the aim of optimizing network performance is discussed in this paper; the authors design a controller to ensure availability of communication resources.

By Jonathan Fink, *Student Member IEEE*, Alejandro Ribeiro, *Member IEEE*, and Vijay Kumar, *Fellow IEEE*

ABSTRACT | In this paper, a system architecture to provide end-to-end network connectivity for autonomous teams of robots is discussed. The core of the proposed system is a cyber–physical controller whose goal is to ensure network connectivity as robots move to accomplish their assigned tasks. Due to channel quality uncertainties inherent to wireless propagation, we adopt a stochastic model where achievable rates are modeled as random variables. The cyber component of the controller determines routing variables that maximize the probability of having a connected network for given positions. The physical component determines feasible robot trajectories that are restricted to safe configurations which ensure these probabilities stay above a minimum reliability level. Local trajectory planning algorithms are proposed for simple environments and leveraged to obtain global planning algorithms to handle complex surroundings. The resulting integrated controllers are robust in that end-to-end communication survives with high probability even if individual point-to-point links are likely to fail with significant probability. Experiments demonstrate that the global planning algorithm succeeds in navigating a complex environment while ensuring that end-to-end communication rates meet or exceed prescribed values within a target failure tolerance.

KEYWORDS | Mobile ad hoc networks; motion planning; multi-robot systems

I. INTRODUCTION

The confluence of advances in wireless communication, sensing, computing, and control has led to the emergence of cyber–physical systems. Cyber–physical systems go beyond the mere addition of cyber functionalities to physical systems being typified by a tight blending of cyber and physical components and functions. A canonical example of this class of systems is the autonomous robot team where a group of mobile robots cooperate to accomplish tasks assigned by human operators. Consider, for example, a search and rescue mission in a hazardous environment where a team of robots is deployed to scout points of interest. Designated lead members of the team move to specified locations while the remaining robots provide mission support by configuring a multihop wireless network that permits relaying of information back to the operators. Availability of wireless communications is critical for task accomplishment because communications are required to exchange information between robots as well as to relay information to and from the human

Manuscript received March 30, 2011; revised June 22, 2011; accepted June 24, 2011. Date of publication September 6, 2011; date of current version December 21, 2011. This work was supported by the U.S. Army Research Laboratory under Grant W911NF-08-2-0004, and the U.S. Office of Naval Research (ONR) under Grants N00014-08-1-0696 and N00014-07-1-0829.

J. Fink was with the GRASP Laboratory, University of Pennsylvania, Philadelphia, PA 19104 USA. He is now with the U.S. Army Research Laboratory, Adelphi, MD 20783-1197 USA (e-mail: jonfink@grasp.upenn.edu).
A. Ribeiro is with the Electrical and Systems Engineering Department, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: aribeiro@seas.upenn.edu).
V. Kumar is with the GRASP Laboratory, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: kumar@grasp.upenn.edu).

Digital Object Identifier: 10.1109/JPROC.2011.2161427

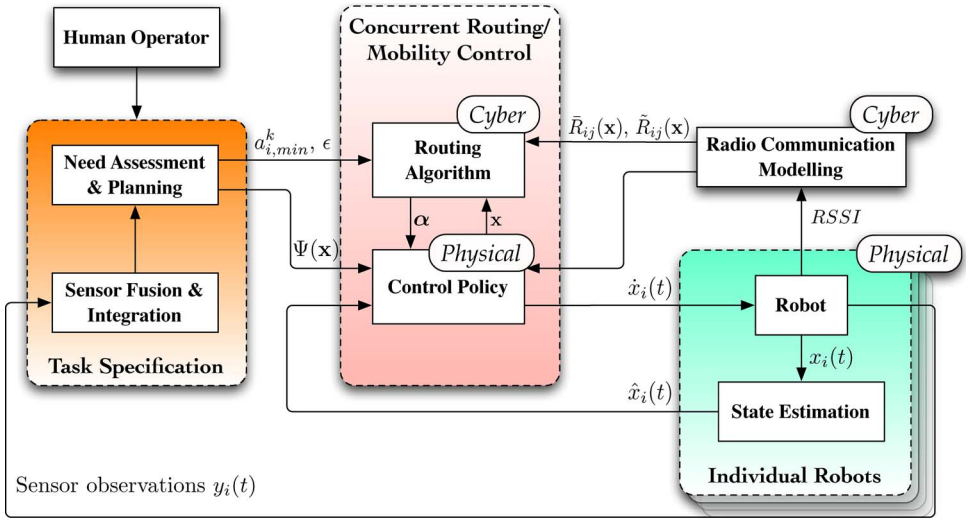

Fig. 1. *System architecture. Task specification here represents a generic spatial application defined by a convex task potential function $\Psi(\mathbf{x})$ while providing a stream of data to the human operator. Individual robot components consist of the low-level robot control, estimation, and communication. We additionally assume that a subsystem is available to build an online model of radio communication in the environment. The focus of this work is on developing concurrent methods for routing and mobility control.*

operators. It then follows that deployment of the team requires integration of (cyber) wireless communications with (physical) mobility control. The cyber–physical perspective posits that this integration is not just a matter of adding wireless transceivers to the robots but requires a tight conjoining of communications and mobility control.

Tight integration of communications and mobility arises in autonomous robot teams because availability of wireless infrastructure is unlikely in the harsh environments in which they are to be deployed. In the absence of infrastructure, robots have to self-organize into a wireless network capable of supporting the necessary information exchanges. In this context, the spatial placement of the robots affects the feasibility of successful communication because some physical configurations cannot support required transmission rates between, say, human operators and some lead members. We can therefore think of communications as determining a set of virtual obstacles that restrict the movement of the robots. However, this is not a hard limit as it is possible to reconfigure communication variables to alter the set of physical configurations for which communications are feasible. This yields a control loop that begins with communication variables limiting mobility and finishes with mobility determining communication variables. In the end, successful deployment of an autonomous team of robots requires joint cyber–physical controllers that determine (physical) trajectories for the robots while ensuring (cyber) availability of communication resources [1].

An architecture diagram for the cyber–physical control of autonomous teams of robots is shown in Fig. 1. As with any mobility control system, there is a block performing task specification, a second block executing the control law, and a third block conducting actuation and state estimation. The task specification block interfaces with the human operators and integrates robot observations and requirements to determine specifications that it passes on to the control block. These specifications come in the form of a potential function $\Psi(\mathbf{x})$ and basal end-to-end communication rate requirements $a^k_{i,\min}$—the subindex i here denotes a source and the superindex k a destination; see Section II. The goal of the autonomous team is to minimize the potential $\Psi(\mathbf{x})$ while ensuring that end-to-end communication rates exceed the minimum required level $a^k_{i,\min}$.

The control block is the core of the cyber–physical system and the main focus of the work described in this paper. The purpose of this block is to determine physical control inputs $\dot{\mathbf{x}}(t)$ that dictate the movement of the robots and network variables $\boldsymbol{\alpha}(t)$ that determine the communication of information through members of the team. The control inputs $\dot{\mathbf{x}}(t)$ are conducive to task completion, i.e., minimization of the task potential, while ensuring reliable network operation, i.e., communication at or exceeding rates $a^k_{i,\min}$ when combined with network variables $\boldsymbol{\alpha}(t)$. Information about achievable rates between different locations is necessary to determine mobility and communication variables. This information is provided by a radio communication modeling block. Due to the difficulty of predicting these rates accurately, the output of this block is in the form of a statistical model of achievable rates. As we discuss in Section III, models that provide the mean and variance of achievable rates are reasonable in practice.

The execution block consists of individual robots implementing the control law $\dot{\mathbf{x}}(t)$ and routing packets

according to variables $\boldsymbol{\alpha}(t)$. Robots also take observations $y_i(t)$, e.g., a video feed, that they relay to task planning and perform position estimation $\hat{x}_i(t)$ that they feedback to the control block. Using available technologies for mapping, control, and state estimation, each robot can estimate its position $\hat{x}_i(t)$ and control its velocity $\dot{x}_i(t)$ with respect to a common known map of the environment. Since we do not provide details about mapping or state estimation in this paper, we refer the interested reader to, e.g., [2].

A. Communication-Aware Mobility Control

In the design of concurrent communications and mobility control algorithms for autonomous robot teams there are three issues that arise naturally: 1) how to translate robot positions into information about the point-to-point rates at which pairs of robots can communicate with each other; 2) how will the cyber–physical control loop in Fig. 1 utilize point-to-point connectivity information in order to ensure end-to-end network integrity; and 3) how to deal with uncertainty in the information about point-to-point channels between individual pairs of robots.

Issues 1) and 2) are related to each other. A simple answer to issue 1) is to identify the ability to communicate with spatial proximity, possibly mixed with line-of-sight information to account for physical obstacles to radio propagation. Using this model, point-to-point links between robots are feasible if they are sufficiently close and within sight of each other and infeasible otherwise. Matched to this simple model of point-to-point connectivity, the ability to establish end-to-end connections is identified with the connectedness of the resulting graph. The problem of ensuring end-to-end network integrity then becomes one of letting robots move to accomplish their task while guaranteeing that the graph of point-to-point connections consists of a single connected component [3]–[10].

The advantage of this approach is that network-wide indicators of connectedness such as the second largest eigenvalue of the graph Laplacian or the k-connectivity of the network's graph are computationally tractable. It is therefore not difficult to insert connectivity constraints into the task planner and, if there is leeway in the selection of different spatial configurations, to find configurations that optimize connectivity in terms of the metric under consideration. Do notice that path planning is computationally challenging even in the absence of communication constraints. The tractability of graph connectedness indicators simply means that the computational complexity of the cyber–physical control loop is comparable to that of a communication-unaware controller.

Aside from the value of simplicity, associating small distance and line-of-sight with connectivity is not entirely justified. Early solutions to wireless networking for systems of dynamic nodes were indeed based on the concept that reliable point-to-point links could be detected and established [11], [12]. However, it was quickly noticed that binary channel models were not accurate representations of wireless links [13]. As a consequence, attention was devoted to methods that explicitly consider the reliability of point-to-point links and choose routes to optimize end-to-end reliability metrics such as probability of delivery [14]–[16]. These more nuanced models of wireless connectivity can be incorporated without much difficulty into mobility control algorithms that use indicators of graph connectedness [17], [18]. If links are characterized by reliability indicators, the binary connectivity graph can be replaced by a weighted graph. Information about the number of connected components can be similarly extracted from, for example, the second largest eigenvalue of the weighted graph's Laplacian.

Either based on binary or weighted graphs, connectedness is an indirect indicator of the ability to establish end-to-end communications. Having a single connected component just implies that multihop paths from sources to destinations exist but does not determine whether the network formed by the robots is able to support desired communication rates. It is possible that some links in the connectivity graph can support rates that are too low, or that the resulting configuration contains bottlenecks that limit information flow. In the cyber–physical control loop depicted in Fig. 1, graph connectedness does not solve for the network variables $\boldsymbol{\alpha}(t)$. Instead, it restricts movement so that a necessary but not sufficient condition for the existence of feasible network variables $\boldsymbol{\alpha}(t)$ is not violated. In fact, one could argue that these controllers are not cyber–physical because the integration of wireless communications with physical mobility is minimal. A true cyber–physical design should consider end-to-end rates and conjoin the design of mobility and network control as done in the systems described in the following section.

B. Robust Cyber–Physical Control

An honest metric of network integrity has to rely on the achievability of target end-to-end communication rates. Communication rates, however, do not depend solely on the spatial configuration but also on the manner in which packets are routed through the network. This leads to the problem of joint cyber–physical control of trajectories and routing variables [1], [19]. For given spatial configuration $\mathbf{x}$, pairs of achievable point-to-point communication rates $R_{ij}(\mathbf{x})$ are assumed available at the radio modeling block. The cyber part of the controller queries the modeling block and attempts to determine routing variables $\boldsymbol{\alpha}(\mathbf{x})$ that support desired communication rates $a^k_{i,\min}$. If this attempt is unsuccessful, spatial configuration $\mathbf{x}$ is marked as unallowable. If the attempt succeeds, joint configuration $(\mathbf{x}, \boldsymbol{\alpha}(\mathbf{x}))$ is deemed feasible and, in particular, spatial configuration $\mathbf{x}$ is deemed allowable. The physical part of the controller utilizes the information on allowable configurations $\mathbf{x}$ to plan physical routes that achieve minimization of the task potential $\Psi(\mathbf{x})$ while ensuring that end-to-end rates exceed basal requirements at all points in time. See Section II.

Ensuring continuous reliable communication is challenging, however, because significant uncertainty in achievable rates is inherent to the deployment of an autonomous robot team. Due to shadowing and small scale fading, even small variations in robots' positions lead to significant changes in channel strength [13], [20], [21] which translate into commensurable changes in achievable communication rates between pairs of robots. Precise channel state information can be acquired through measurements over time, but mobility planning algorithms necessitate access to channel quality indicators at future positions to which the robots are yet to be deployed. Thus, from the perspective of the system in Fig. 1, rates $R_{ij}(\mathbf{x})$ are random. The outputs of the radio modeling block are rate estimates $\bar{R}_{ij}(\mathbf{x})$ whose variances $\tilde{R}_{ij}(\mathbf{x})$ are typically large. See Section III.

Since achievability of end-to-end rates cannot be guaranteed because of rate uncertainties, it is natural to redefine network survivability in terms of the probability of end-to-end rates exceeding their basal rates $a^k_{i,\min}$. Packet routing variables $\boldsymbol{\alpha}(\mathbf{x})$ are then determined in order to maximize this probability [1]. The resulting routes are robust because they are chosen to minimize the probability of failure. At this point it has to be noted that the goal of the self-organized network is to maintain reliable end-to-end communication, not point-to-point communication. Consequently, we can think of robust routes as exploiting spatial redundancy to minimize the effect of point-to-point uncertainty in end-to-end communication rates. Indeed, by splitting traffic flows between various neighboring robots, we can ensure that while failure of a particular link may reduce end-to-end communication rates, it does not interrupt them completely. See Section IV.

The integration of robust routing with the physical control loop is discussed in Section V. We develop local planning algorithms for simple environments (Section V-A) that we test in simulations for teams composed of three and four robots (Section V-B). These simulations demonstrate that the performance of local controllers is hindered by local minima. To overcome this problem as well as to handle complex environments, we leverage the local controllers to design global planning algorithms (Section V-C). Global planners are tested with experimental results in an indoor office space at the University of Pennsylvania (Section V-D). The experiments demonstrate that the global planning algorithm succeeds in navigating a complex environment while ensuring that end-to-end communication rates meet or exceed prescribed values within a target failure tolerance. We close the paper with concluding remarks (Section VI).

II. COMMUNICATIONS AND MOBILITY

Consider a team of N robots and denote their positions as x_i, for $i = 1, \ldots, N$. The robots are kinematic and fully controllable which allows us to consider simple mobility models of the form $\dot{x}_i(t) = u_i(t)$, where $u_i(t)$ is the control input to robot i. A human operator is located at the fixed operation center that we index as $i = 0$ at position x_0. Further define vectors $\mathbf{x} := (x_0, \ldots, x_N) \in \mathbb{R}^{2(N+1)}$ and $\dot{\mathbf{x}} := (\dot{x}_0, \ldots, \dot{x}_N) \in \mathbb{R}^{2(N+1)}$ to group all positions and velocities, respectively. The task assigned to the team is specified through a generic scalar convex task potential function $\Psi : \mathbb{R}^{2(N+1)} \to \mathbb{R}$. If the potential minimum $\Psi_{\min}$ is attained at $\mathbf{x}^*$, i.e., if $\Psi(\mathbf{x}^*) = \Psi_{\min}$, the configuration $\mathbf{x}^*$ satisfies task completion. For example, if a designated leader agent ℓ must visit a target location $x_{\ell,\text{goal}} \in \mathbb{R}^2$, we can define $\Psi(\mathbf{x}) = \|x_\ell - x_{\ell,\text{goal}}\|^2$. The minimum $\Psi_{\min} = 0$ is attained by any configuration $\mathbf{x}^* = (x_0, \ldots, x_\ell, \ldots, x_N)$ for which $x_\ell = x_{\ell,\text{goal}}$, or equivalently by any member of the set $\mathbf{x}^* \in \{\mathbf{x} = (x_0, \ldots, x_\ell, \ldots, x_N) : x_\ell = x_{\ell,\text{goal}}\}$. Irrespective of the particular form of $\Psi(\mathbf{x})$, the control problem is to find velocities $\dot{\mathbf{x}}(t)$ such that at some time t_f the team configuration $\mathbf{x}(t_f) = \mathbf{x}(0) + \int_0^{t_f} \dot{\mathbf{x}}(t)dt$ satisfies task completion in that we have $\Psi(\mathbf{x}(t_f)) = \Psi_{\min}$. Mathematically, we can write this mobility control formulation as

$$\min_{\dot{\mathbf{x}}(t), t\in[0,t_f]} \quad \Psi\big(\mathbf{x}(t_f)\big)$$
$$\text{subject to} \quad \mathbf{x}(t) = \mathbf{x}(0) + \int_0^t \dot{\mathbf{x}}(u)\, du. \tag{1}$$

As robots move to accomplish their task, they maintain *end-to-end* data communication flows between members of the team and/or members of the team and the operation center. Information flows are indexed as $k = 1, \ldots, K$. Flows may have multiple sources and multiple destinations. The set of destinations of the kth information flow is denoted as $\text{dest}(k)$. For agent i and flow k, the variable $a^k_{i,\min}$ represents the required communication rate between agent i and any of the agents in the set of destinations $\text{dest}(k)$. For example, if the only communications of interest are from the lead robot ℓ to the operation center, there are only $K = 1$ flows. Since the flow $k = 1$ is intended to the operating center, $\text{dest}(1) = 0$ and $a^1_{\ell,\min}$ denotes the minimum level of service for the communication from the leader to the operating center. All other variables $a^k_{i,\min} = 0$ are null.

We model *point-to-point* connectivity through a rate function $R_{ij}(\mathbf{x}) = R_{ij}(x_i, x_j)$ that determines the amount of information that agent i at position x_i can send to agent j at position x_j. Since direct communication between the source and the destination of an information flow is not always possible, terminals self-organize into a multihop network to relay packets for each other. Packet relaying is determined by routing variables α^k_{ij} which describe the fraction of time node i spends transmitting data for flow k to node j; see Fig. 2. Thus, the product $\alpha^k_{ij} R_{ij}(\mathbf{x})$ determines the rate of point-to-point information transmission from

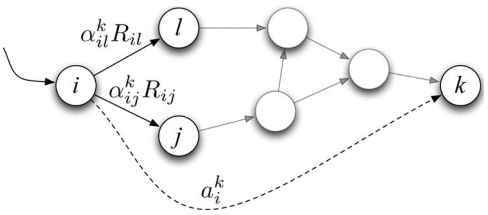

Fig. 2. *Communication network. The nodes are deployed to support end-to-end rates from node i to destination (or flow) k. Routing variables α_{ij}^k determine the fraction of time node i sends packets to node j for flow k. R_{ij} is defined as the supported rate of the wireless channel from node i to node j.*

i to j. If we consider the transmission to all neighboring terminals for which $R_{ij}(\mathbf{x}) > 0$, the total rate at which packets leave agent i is $\sum_{j=0}^{N} \alpha_{ij}^k R_{ij}(\mathbf{x})$. Likewise, the total rate at which i receives packets from other terminals is $\sum_{j=0, j\notin \mathrm{dest}(k)}^{N} \alpha_{ji}^k R_{ji}(\mathbf{x})$. The information rate $a_i^k(\boldsymbol{\alpha}, \mathbf{x})$ available for flow k at source i is the difference between outgoing and incoming rates

$$a_i^k(\boldsymbol{\alpha}, \mathbf{x}) = \underbrace{\sum_{j=0}^{N} \alpha_{ij}^k R_{ij}(\mathbf{x})}_{\text{Outgoing packets}} - \underbrace{\sum_{j=0, j\notin \mathrm{dest}(k)}^{N} \alpha_{ji}^k R_{ji}(\mathbf{x})}_{\text{Incoming packets}} \tag{2}$$

where we define the vector $\boldsymbol{\alpha}$ grouping all routing variables α_{ij}^k. Notice that the variables α_{ij}^k represent time slot shares and must therefore satisfy $0 \le \alpha_{ij}^k \le 1$ for all i, j, and k. It must also be that $\sum_{j,k} \alpha_{ij}^k \le 1$ for all i to ensure that the sum of all time shares at terminal i does not exceed 1. It is possible to, alternatively, require $\sum_{i,j,k} \alpha_{ij}^k \le 1$ if we wish for only one link to be active at any time across the entire network.

Routing variables $\boldsymbol{\alpha}$ and configuration-dependent rates $R_{ij}(\mathbf{x})$ determine the set $a_i^k(\boldsymbol{\alpha}, \mathbf{x})$ of end-to-end communication rates from each node i and flow k as per (2). The task specification requires that end-to-end rates exceed the minimum threshold $a_{i,\min}^k$. Therefore, integrity of the communication network necessitates that

$$a_i^k(\boldsymbol{\alpha}, \mathbf{x}) \ge a_{i,\min}^k, \qquad \text{for all } i, k. \tag{3}$$

Notice that $a_i^k(\boldsymbol{\alpha}, \mathbf{x})$ is a function of positions $\mathbf{x}$ and routing variables $\boldsymbol{\alpha}$. To control end-to-end connectivity, i.e., to satisfy (3), we can resort to control of positions $\mathbf{x}$, to control of routes $\boldsymbol{\alpha}$, or both.

Since communication is necessary for task completion, the mobility control problem as summarized in (1) is redefined. The new goal is to find algorithms and control policies that govern robot motions in order to satisfy the task specifications in (1) and (3). Reducing $\Psi(\mathbf{x})$ as per (1) and ensuring network integrity as per (3) may be conflicting requirements. We therefore replace (1) by a concurrent search of trajectories $\mathbf{x}(t)$ and routes $\boldsymbol{\alpha}(t)$ so that the task potential is minimized without ever breaking communication connectivity. Mathematically, we write this objective as the optimization problem

$$\begin{aligned} \min_{\boldsymbol{\alpha}(t), \dot{\mathbf{x}}(t), t\in[0,t_f]} \quad & \Psi\big(\mathbf{x}(t_f)\big) \\ \text{subject to} \quad & a_i^k(\boldsymbol{\alpha}(t), \mathbf{x}(t)) \ge a_{i,\min}^k \\ & \mathbf{x}(t) = \mathbf{x}(0) + \int_0^t \dot{\mathbf{x}}(u)\, du \end{aligned} \tag{4}$$

where rates $a_i^k(\boldsymbol{\alpha}(t), \mathbf{x}(t))$ are given by the expression in (2) with $\boldsymbol{\alpha} = \boldsymbol{\alpha}(t)$ and $\mathbf{x} = \mathbf{x}(t)$.

A drawback of the formulation in (4) is the difficulty of ensuring that the constraints in (3) are satisfied. As seen in (2), rates $a_i^k(\boldsymbol{\alpha}, \mathbf{x})$ depend on the link reliabilities $R_{ij}(\mathbf{x})$, which are difficult to estimate. Propagation knowledge is used by the radio communication modeling block in Fig. 1 to provide rough rate estimates $\bar{R}_{ij}(\mathbf{x})$. Agents can further measure received signal strength indicators (RSSIs) to refine the rate estimates for the current configuration $\mathbf{x}(t)$; see, e.g., [22]. The challenge here is that $\bar{R}_{ij}(\mathbf{x})$ estimates are needed not only for configuration $\mathbf{x}(t)$, but for nearby configurations to which the robots may move. The high variability of wireless channels makes $\bar{R}_{ij}(\mathbf{x}(t))$ a poor predictor of $R_{ij}(\mathbf{x})$ even if $\mathbf{x}$ is close to $\mathbf{x}(t)$. In general, channel estimates $\bar{R}_{ij}(\mathbf{x})$ at future positions $\mathbf{x}$ follow from radio propagation modeling and corrections from RSSI observations.

In formal terms, uncertainty of channel estimates means that the variances $\tilde{R}_{ij}(\mathbf{x})$ of channel estimates $\bar{R}_{ij}(\mathbf{x})$ are typically large for possible future positions $\mathbf{x}$. Variances $\tilde{R}_{ij}(\mathbf{x})$ can be provided by the radio modeling block and are therefore assumed to be available at the controller along with the mean estimates $\bar{R}_{ij}(\mathbf{x})$. We seek to redefine (4) in a manner that takes into account this probabilistic formulation of channel rates. The important observation here is that if point-to-point link rates become random, so do the rates $a_i^k(\boldsymbol{\alpha}, \mathbf{x})$ of end-to-end communication flows [cf. (2)]. Consequently, it is not possible to guarantee satisfaction of the constraints in (3). Rather, we introduce a reliability tolerance ϵ and require that for all i and k

$$\mathrm{P}\left[a_i^k(\boldsymbol{\alpha}, \mathbf{x}) \ge a_{i,\min}^k\right] \ge \epsilon. \tag{5}$$

That is, we require that the end-to-end link between all sources i and the destinations of all corresponding flows k

exceed their minimum required level of service with probability larger than ϵ. As in the case of the rate requirement in (3), we can satisfy (5) by controlling $\boldsymbol{\alpha}$ and $\mathbf{x}$ separately or jointly.

In order to robustly satisfy the networking constraints in (5), the concurrent routing and mobility problem (4) is replaced by

$$\begin{aligned} \min_{\boldsymbol{\alpha}(t),\dot{\mathbf{x}}(t),t\in[0,t_f]} \quad & \Psi\big(\mathbf{x}(t_f)\big) \\ \text{subject to} \quad & \mathrm{P}\Big[a_i^k(\boldsymbol{\alpha},\mathbf{x}) \geq a_{i,\min}^k\Big] \geq \epsilon \\ & \mathbf{x}(t) = \mathbf{x}(0) + \int_0^t \dot{\mathbf{x}}(u)\, du. \end{aligned} \tag{6}$$

The focus of this paper is the solution of (6).

The problem formulation in (6) inherits some standard complications from the control formulation in (1). The problem is infinite dimensional and due to, e.g., the presence of obstacles, not convex. The concurrent search in (6) is further complicated by the entanglement of the routing and mobility problems. We deal with this entanglement by fixing $\mathbf{x}$ and selecting $\boldsymbol{\alpha}$ in a manner that optimizes the reliability $\mathbf{P}[a_i^k(\boldsymbol{\alpha},\mathbf{x}) \geq a_{i,\min}^k]$ (Section IV). We finish with local and global searches on positions $\mathbf{x}$ to minimize $\Psi(\mathbf{x})$ while keeping reliabilities above the ϵ threshold (Section V).

III. POINT-TO-POINT RATE MODELING

With robots i and j located at positions x_i and x_j, we seek to develop a probabilistic model for the supported communication rate $R_{ij}(\mathbf{x}) = R_{ij}(x_i, x_j)$ between them. More specifically, we wish to model its expected value $\bar{R}_{ij}(\mathbf{x}) = \bar{R}_{ij}(x_i, x_j)$ and variance $\tilde{R}_{ij}(\mathbf{x}) = \tilde{R}_{ij}(x_i, x_j)$. Our intent is to use simple radios that do not perform rate or power adaptation—we use 2.4-GHz Zigbee radios in our experiments; see Section V-D. In that case, the communication rate $R_{ij}(x_i, x_j)$ is a function of the packet error rate of the channel, which in turn is a function of the signal-to-noise ratio [23]. We therefore focus on models of the received signal strength $P_R(x_i, x_j)$ that we cascade into models of the packet error rate and the supported communication rate $R_{ij}(x_i, x_j)$.

Received signal power $P_R(x_i, x_j)$ is determined by three phenomena: path loss due to the distance from the source, shadowing due to obstacles in the propagation path, and multipath fading that arises as a result of reflections and refractions. Of these three phenomena, path loss and shadowing can be incorporated into a predictive model with relative ease. Fading, however, is difficult to predetermine. Multipath fading arises due to a spatial wave interference pattern generated by reflections and refractions of the electromagnetic wave which, as a consequence, inherits a space constant in the order of the wave's wavelength. Thus, in order to use ray tracing models to predict the interference pattern we need to locate all potential reflectors and refractors with precision smaller than this wavelength. While plausible in principle, this is intractable in practice because wavelengths at common operating frequencies are too small—for example, with radios operating at 2.4 GHz., the wavelength is about 12 cm. Given this difficulty we adopt the following model for the received power $P_{R,\mathrm{dBm}}(x_i, x_j) = 10\log(P_R(x_i, x_j))$ measured in dBm:

$$\begin{aligned} &P_{R,\mathrm{dBm}}(x_i, x_j) \\ &\quad = \underbrace{L_0 - 10n \cdot \log\big(\|x_i - x_j\|\big)}_{\text{Path loss}} - \underbrace{W(x_i, x_j)}_{\text{Shadowing}} - \underbrace{\mathcal{F}}_{\text{Fading}} \end{aligned} \tag{7}$$

where the term $\mathcal{F}$ is a zero-mean Gaussian random variable with variance $\sigma_{\mathcal{F}}^2$ modeling fading effects. The term L_0 is the measured power at a reference distance d_0 from the source, n is a path loss exponent, and $W(x_i, x_j)$ is a non-smooth function to model shadowing as a function of the number of obstacles between source and destination. Experimental data collected in an indoor office environment at the University of Pennsylvania along with the function in (7) having parameters fit to the experimental data are depicted in Fig. 3. The parameters in Fig. 3 are $L_0 = -51$ dBm for $d_0 = 1$ m, $n = 2.1$, $W(x_i, x_j) = 0$ for line-of-sight links and $W(x_i, x_j) = 7.6$ dB for non-line-of-sight links, and $\sigma_{\mathcal{F}}^2 = 32$ dB^2. Notice that fading can cause

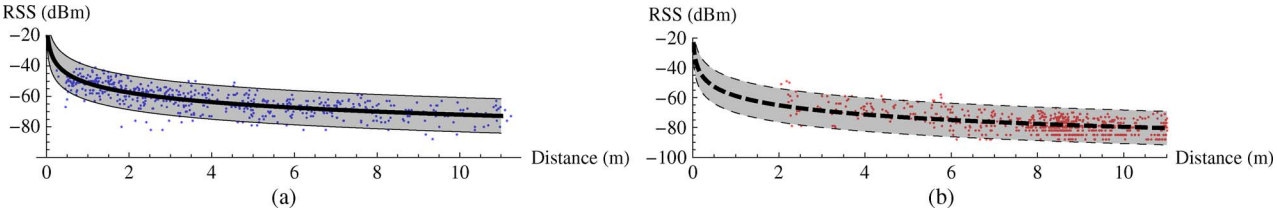

Fig. 3. *Experimental characterization of received power model (7) based on 1000 samples when we classify wireless channels based on line-of-sight. (a) Line-of-sight. (b) Non-line-of-sight.*

variations in received power in the order of ± 10 dB. This is verified by the standard deviation of $\mathcal{F}$, which is $\sigma_{\mathcal{F}} \approx 5.7$ dB. Recall that a 10-dB difference corresponds to a change of an order of magnitude for the power measured in linear units.

Translation of received signal strength $P_R(x_i, x_j)$ into packet error rate $p_e(P_R(x_i, x_j))$ depends on the type of modulation and the choice of error correcting codes. Regardless of specifics, a generally good approximation to the packet error rate is

$$p_e\big(P_R(x_i, x_j)\big) = \operatorname{erfc}\left(\sqrt{k\frac{P(x_i, x_j)}{P_{N_0}}}\right) \tag{8}$$

where P_{N_0} is the noise power, $\operatorname{erfc}(x)$ is the complementary error function, and k is a constant that depends on modulation and coding [23]. The translation into communication rates $R_{ij}(x_i, x_j)$ is now straightforward as it simply requires multiplying the information rate of transmitted packets R_0 by the probability of successful decoding, i.e., $R_{ij}(x_i, x_j) = R_0[1 - p_e(P_R(x_i, x_j))]$.

The randomness of the fading term $\mathcal{F}$ in (7) propagates into randomness in the error probabilities $p_e(P(x_i, x_j))$ and subsequently into randomness in the rates $R_{ij}(x_i, x_j)$. To compute the mean and variance of $R_{ij}(x_i, x_j)$ we need to recall that $\mathcal{F}$ is a Gaussian random variable. We can then invoke the delta method [24] to approximate the mean and variance of $R_{ij}(x_i, x_j)$ in terms of the mean and variance of $P_R(x_i, x_j)$ and the derivatives of the function $p_e(P_R(x_i, x_j))$ in (8). The complete mapping from distance to link rates is depicted in Fig. 4. Observe that except for very short distances smaller than 3 m there is significant variability in achieved link rates. See [1, ch. 4] for details.

While the model we present in this section describes the statistics of the supported rate $R_{ij}(x_i, x_j)$ in terms of distances $\|x_i - x_j\|$, we emphasize that this is not a limitation of the proposed cyber–physical controller. In fact, the controller is agnostic to the particulars of the radio propagation model and relies only on the ability to query for means and variances. It is reasonable to expect that techniques for spatial mapping of point-to-point communication rates will outperform distance-based models when considering unknown or highly dynamic environments. However, as demonstrated by the experiments in Section V-D, the robust satisfaction of communication constraints affords the possibility to rely on coarse models of point-to-point communication rates while still ensuring maintenance of quality-of-service requirements.

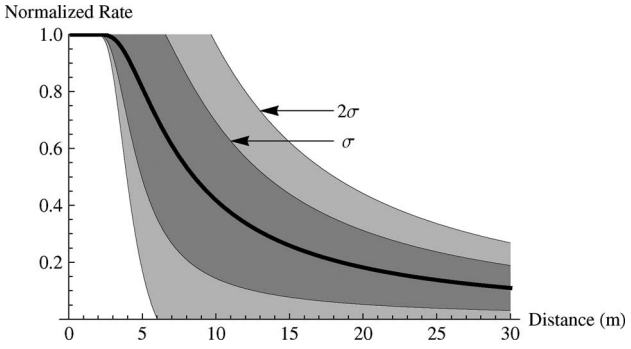

Fig. 4. *Mapping from distances $\|x_i - x_j\|$ between a transmitter at location x_i and a receiver at position x_j to normalized link rates $R_{ij}(x_i, x_j)/R_0$.*

IV. ROBUST ROUTING

The control loop in Fig. 1 embeds a cyber optimization component and a physical reconfiguration module. The cyber optimization component optimizes communication routes for given positions, while the physical component explores spatial reconfigurations that maintain network integrity and are conducive to task achievement. The physical module relies on the cyber module for the determination of configurations that are feasible from the perspective of maintaining target communication rates with some desired reliability.

The major difficulty in designing the cyber component of this control loop is the uncertainty in achievable transmission rates between nearby agents. As we discussed in Section III, simple experiments suffice to demonstrate that achievable rates in wireless links are difficult to predict. Therefore, assuming that the actual channel rates $R_{ij}(\mathbf{x})$ coincide with their estimates $\bar{R}_{ij}(\mathbf{x})$ may result in a drastic difference between predicted and actual end-to-end rates. This mismatch subsequently leads to situations in which the communication optimization module asserts network integrity when actual rates do not suffice to support the task assigned by operators.

A simple way to account for the uncertainty in $R_{ij}(\mathbf{x})$ is to discount $\bar{R}_{ij}(\mathbf{x})$ in order to reduce the likelihood of having actual rates smaller than the assumed value. This is definitely possible but would result in underutilization of communication resources. A better way to account for the uncertainty in channel rates is to recall that end-to-end rather than point-to-point failures are relevant. With reference to Fig. 2, say that robot i decides to convey information to k by forwarding it through l. If the rate $R_{il}(\mathbf{x})$ happens to be substantially smaller than estimated, communication is interrupted. If, on the contrary, robot i decides to send part of the information through l and some other part through j, the communication between i and k survives as long as the rate $R_{ij}(\mathbf{x})$ is larger than estimated. This example illustrates that it is possible to exploit spatial redundancy through traffic splitting in order to devise robust routes that guarantee small changes in end-to-end rates despite large variability in point-to-point rates $R_{ij}(\mathbf{x})$ [25].

To develop robust routing algorithms, start by noticing that computing the probability in (5), which is part of the

problem formulation in (6), necessitates modeling the probability distribution of $a_i^k(\boldsymbol{\alpha}, \mathbf{x})$. This is difficult in general. However, if we explicitly consider the stochastic model of point-to-point links via their means and variances, we can compute the mean and variance of end-to-end rates $a_i^k(\boldsymbol{\alpha}, \mathbf{x})$ as

$$\begin{aligned}\bar{a}_i^k(\boldsymbol{\alpha}, \mathbf{x}) &:= \mathbb{E}\left[a_i^k(\boldsymbol{\alpha}, \mathbf{x})\right] \\ &= \sum_j \alpha_{ij}^k \bar{R}_{ij}(\mathbf{x}) - \sum_{j \notin \mathrm{dest}(k)} \alpha_{ji}^k \bar{R}_{ij}(\mathbf{x}) \qquad (9)\\ \tilde{a}_i^k(\boldsymbol{\alpha}, \mathbf{x}) &:= \mathrm{var}\left[a_i^k(\boldsymbol{\alpha}, \mathbf{x})\right] \\ &= \sum_j \left(\alpha_{ij}^k\right)^2 \tilde{R}_{ij}(\mathbf{x}) + \sum_{j \notin \mathrm{dest}(k)} \left(\alpha_{ji}^k\right)^2 \tilde{R}_{ij}(\mathbf{x}). \qquad (10)\end{aligned}$$

A proxy for the probability in (5) is the difference between $a_i^k(\boldsymbol{\alpha}, \mathbf{x})$ and its mean $\bar{a}_i^k(\boldsymbol{\alpha}, \mathbf{x})$ normalized by its standard deviation $\sqrt{\tilde{a}_i^k(\boldsymbol{\alpha}, \mathbf{x})}$. Indeed, a typical approximation for the probability constraint in (5) is the condition

$$P_i^k(\boldsymbol{\alpha}, \mathbf{x}) := \frac{\bar{a}_i^k(\boldsymbol{\alpha}, \mathbf{x}) - a_{i,\min}^k}{\sqrt{\tilde{a}_i^k(\boldsymbol{\alpha}, \mathbf{x})}} \geq \Phi^{-1}(\epsilon) \qquad (11)$$

for some function $\Phi^{-1}(\epsilon)$. If the probability distribution of rates $R_{ij}(\mathbf{x})$ is Gaussian, (11) is equivalent to (5) if $\Phi^{-1}(\epsilon)$ is the inverse of the normal distribution's cumulative distribution function. For other probability distributions, we can apply Chebyshev's inequality with $\Phi^{-1}(\epsilon) = \sqrt{1/\epsilon}$ to demonstrate that (11) is a sufficient condition for satisfying (5) though this will usually be a conservative approximation.

For given positions $\mathbf{x}$, the goal of the cyber optimization component is to find routing variables $\boldsymbol{\alpha}$ that satisfy (11) which is equivalent to, approximate to, or a sufficient condition to satisfy (5). In either case, there is some indeterminacy because there is a nonunique set of variables $\boldsymbol{\alpha}$ that satisfy the corresponding inequality. This indeterminacy provides a degree of freedom that can be used to increase the reliability beyond the required level. For a given configuration $\mathbf{x}$, we would like to find routes $\boldsymbol{\alpha} = \boldsymbol{\alpha}(\mathbf{x})$ that provide the maximum possible reliability. Doing so is not conceptually difficult—it requires determination of routes $\boldsymbol{\alpha}(\mathbf{x})$ that maximize the probability proxies $P_i^k(\boldsymbol{\alpha}, \mathbf{x})$ in (11) in some sense. Considering that there is a rate variable a_i^k for each source–destination pair, there are several possible choices to define this maximization. One possibility is to maximize the sum of these probability proxies. Another possibility is to select routes $\boldsymbol{\alpha}(\mathbf{x})$ that make the smallest of these terms as large as possible. Maximizing this minimum implies that the constraints in (11) are satisfied with significant slack and that there is significant liberty to change the physical configuration without violating communication constraints. This freedom of movement facilitates implementation of the physical mobility control block as we discuss in Section V.

An important observation here is that the probability proxy constraints in (11) can be written in a manner that defines a cone in the joint space of routing variables and end-to-end rates [1, ch. 5]. A consequence of this property is that the optimization problems that need to be solved to maximize them are second-order cone program (SOCPs). SOCPs are a particular class of convex optimization problem that can be solved by efficient polynomial-time algorithms. For the problems considered here, the computational complexity of these algorithms is represented as a polynomial function of the number of agents N and the number of flows (destinations) K, as $O((K \cdot N^2)^{3.5})$. In practical implementations, the N^2 term can be reduced by eliminating links for which rate estimates $\bar{R}_{ij}(\mathbf{x})$ are below a certain threshold.

Recall that the motivation for robust routing algorithms is to reduce uncertainty in end-to-end communication rates. We do so by taking advantage of spatial redundancy, for which it is necessary to split traffic among various different routes. It is fitting that we do expect to obtain this type of solution from the maximization of the probability proxies $P_i^k(\boldsymbol{\alpha}, \mathbf{x})$ in (11). Indeed, increases in these terms can be brought about by either increasing the mean $\bar{a}_i^k(\boldsymbol{\alpha}, \mathbf{x})$ or decreasing the variance $\tilde{a}_i^k(\boldsymbol{\alpha}, \mathbf{x})$. Since the mean is a linear function of $\boldsymbol{\alpha}$, traffic splitting has a minor effect on $\bar{a}_i^k(\boldsymbol{\alpha}, \mathbf{x})$. However, traffic splitting reduces $\tilde{a}_i^k(\boldsymbol{\alpha}, \mathbf{x})$ by a factor proportional to the splitting because the variance is a quadratic function of $\boldsymbol{\alpha}$—recall that $\alpha_{ij} \leq 1$. Thus, traffic splitting tends to increase the probability proxies because it keeps $\bar{a}_i^k(\boldsymbol{\alpha}, \mathbf{x})$ more or less constant and reduces $\tilde{a}_i^k(\boldsymbol{\alpha}, \mathbf{x})$ significantly.

V. CONTROL

As per (6), the objective of mobility control is to decrease $\Psi(\mathbf{x})$ while satisfying the probability proxy constraints in (11)—which are equivalent or approximations to $\mathbf{P}[a_i^k(\boldsymbol{\alpha}, \mathbf{x}) \geq a_{i,\min}^k] \geq \epsilon$. To check for the feasibility of a *cyber–physical* configuration $(\boldsymbol{\alpha}, \mathbf{x})$, we define the probability margin as the minimum slack in probability proxy constraints across all flows and sources

$$\nu(\boldsymbol{\alpha}, \mathbf{x}) := \min_{i,k} \left[\frac{\bar{a}_i^k(\boldsymbol{\alpha}, \mathbf{x}) - a_{i,\min}^k}{\sqrt{\tilde{a}_i^k(\boldsymbol{\alpha}, \mathbf{x}')}} - \Phi^{-1}(\epsilon) \right]. \qquad (12)$$

Notice that a necessary and sufficient condition for feasibility of the *physical* configuration $\mathbf{x}$ is to have $\nu(\boldsymbol{\alpha}(\mathbf{x}), \mathbf{x}) \geq 0$, with routing variables $\boldsymbol{\alpha}(\mathbf{x})$ as given by the solution of the SOCP described in Section IV. A

sufficient condition for feasibility of physical configuration $\mathbf{x}'$ is the existence of a cyber configuration $\boldsymbol{\alpha}$ for which $\nu(\boldsymbol{\alpha}, \mathbf{x}') \geq 0$. In particular, for $\mathbf{x}'$ close to $\mathbf{x}$, we expect to have $\nu(\boldsymbol{\alpha}(\mathbf{x}), \mathbf{x}') \geq 0$ since the channel statistics at $\mathbf{x}$ and $\mathbf{x}'$ are close.

A. Gradient Controller

In general, gradient controllers define velocities $\dot{\mathbf{x}}(t)$ proportional to the negative gradients $-\nabla\Psi(\mathbf{x}(t))$ of the task potential. Since the problem in (6) is subject to communication constraints, a local controller will be based on potential gradients projected onto the feasible set $\nu(\boldsymbol{\alpha}, \mathbf{x}) \geq 0$. The complex description of the feasible set, however, precludes computation of projected gradients. Instead, we consider the probability margin $\nu(\boldsymbol{\alpha}(\mathbf{x}), \mathbf{x})$ and modify the potential $\Psi(\mathbf{x})$ by adding the probability margin constraint into the objective through a barrier function

$$\Omega(\mathbf{x}) := \Psi(\mathbf{x}) - \log(\nu(\boldsymbol{\alpha}(\mathbf{x}), \mathbf{x})). \tag{13}$$

Since nonnegativity is necessary and sufficient for feasibility of physical configuration $\mathbf{x}$, the potential $\Omega(\mathbf{x})$ in (13) is defined if and only if physical configuration $\mathbf{x}$ is feasible.

The local control law is defined to implement gradient descent on the modified potential $\Omega(\mathbf{x})$ introduced in (13), which, in explicit terms, is given by

$$\mathbf{u}(t) = -\nabla\Psi(\mathbf{x}(t)) + \frac{\nabla_{\mathbf{x}}\nu[\boldsymbol{\alpha}(\mathbf{x}(t)), \mathbf{x}(t)]}{\nu[\boldsymbol{\alpha}(\mathbf{x}(t)), \mathbf{x}(t)]}. \tag{14}$$

The term $\nabla\Psi(\mathbf{x}(t))$ in (14) drives the system to satisfy the task potential. The term $\nabla_{\mathbf{x}}\nu[\boldsymbol{\alpha}(\mathbf{x}(t)), \mathbf{x}(t)]/\nu[\boldsymbol{\alpha}(\mathbf{x}(t)), \mathbf{x}(t)]$ serves as a barrier that drives robots away from configurations for which there is a low probability of exceeding the desired reliability in end-to-end rates.

B. Simulation Results

We implement the local controller with mobility control inputs given by (14) and communication variables obtained from Section IV. Computing controls based on local optimization of the network-level end-to-end rates allows for a method of realizing team deployment while maintaining the necessary level of network connectivity. Fig. 5 depicts an example deployment with three robots for a time-varying task potential

$$\Psi(\mathbf{x}(t)) = \begin{cases} x_{2,\text{goal}} = (4, 0), & t < 40 \\ x_{2,\text{goal}} = (8, 0), & t < 60 \\ x_{2,\text{goal}} = (6, 5), & t \geq 60. \end{cases} \tag{15}$$

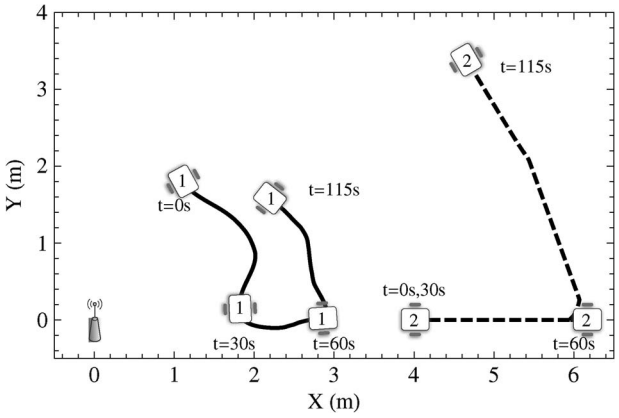

Fig. 5. *Deployment via local control law (14) for a system with a fixed access point, relay node x_1 and lead node x_2, which is controlled by a time-varying task potential $\Psi(\mathbf{x}(t))$.*

Most importantly, this example demonstrates convergence of the task potential $\Psi(\mathbf{x})$ while maintaining $\mathbf{P}[a_i \geq a_{i,\min}] > \epsilon$ as depicted in Fig. 6(a) and (c). However, it is also interesting to observe that when the task potential is minimized, e.g., $t \leq 30$ s, the local control law (14) maximizes the probability of each end-to-end rate exceeding its minimum threshold. When the task potential switches so that $\Psi(\mathbf{x})$ is no longer minimized, the probability margin is reduced so that the primary objective, minimization of $\Psi(\mathbf{x})$, is prioritized. Finally, Fig. 6(b) depicts the end-to-end rate of the node x_2 that must remain above $a_{2,\min} = 0.1$. Remember that $R_{ij}(\mathbf{x})$ is a stochastic rate that affects the end-to-end rate. The envelope around $\bar{a}_2$ in Fig. 6(b) depicts the effect that different realizations of communication channels $R_{ij}(\mathbf{x})$ will have on the end-to-end rate. Since the pursuit of minimization on $\Psi(\mathbf{x})$ is constrained to have a probability margin $\nu(\boldsymbol{\alpha}(\mathbf{x}), \mathbf{x}) > 0$, the end-to-end rate exceeds its threshold in the presence of deviations to $R_{ij}(\mathbf{x})$.

We perform a four robot simulation, depicted in Fig. 7, to demonstrate how the complexity of the objective function (13) increases with more nodes. Similar to the three robot simulation above, the task potential function is time varying so that $\Psi(\mathbf{x}) = 0$ for $t < 50$ s and $\Psi(\mathbf{x}) = \|x_3 - (9, 0)\|$ for $t \geq 50$ s. When $t < 40$ and the task potential is already minimized, control of x_1 and x_2 is based solely on the maximization of $\log \nu(\boldsymbol{\alpha}(\mathbf{x}), \mathbf{x})$. From the symmetric initial configuration, the maximization drives x_1 and x_2 towards a local maxima where they would be positioned at the same point. In fact, if we consider all configurations where $x_1 < x_2$ and both agents are on the line connecting the access point, we find that there are two local maxima as depicted in Fig. 8.

To further illustrate the virtual terrain of the objective function (13) that drives local control, we observe the convergence of $\Psi(\mathbf{x}(t))$ as depicted in Fig. 9(a). For times 70 s $< t <$ 90 s, convergence of $\Psi(\mathbf{x}(t))$ slows,

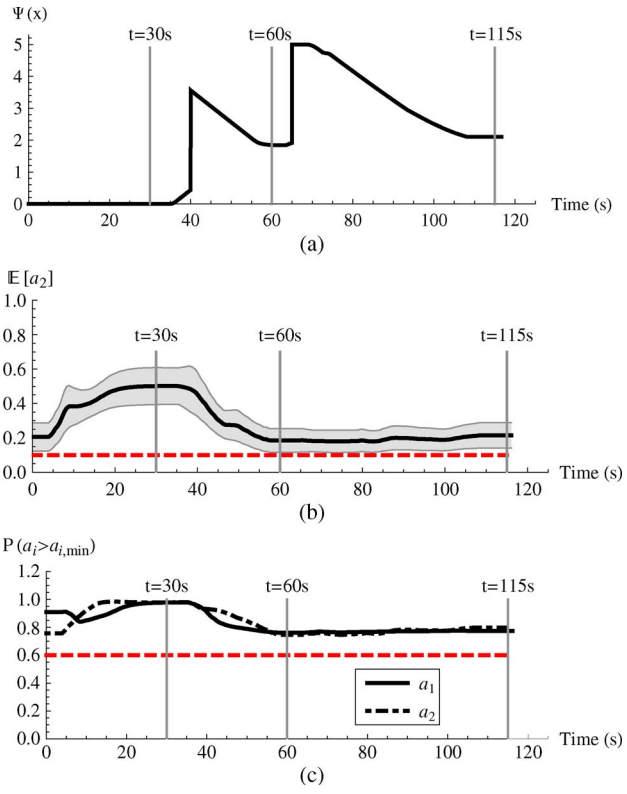

Fig. 6. ***Performance of the local control law (14) demonstrating convergence of $\Psi(\mathbf{x}(t))$ in (a), the maintenance of expected end-to-end rate greater than the threshold of $a_{2,\min} = 0.1$ in (b), and the $\mathbf{P}[a_i \geq \alpha_{i,\min}] > 0.6$. The envelope surrounding $\mathbb{E}[a_2]$ in (b) depicts the 60% confidence interval for realizations of the end-to-end rate with stochastic $R_{ij}(\mathbf{x})$.***

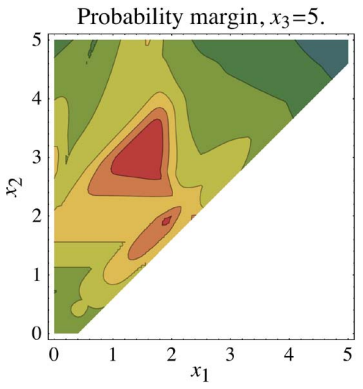

Fig. 8. ***Evaluation of $\nu(\alpha(\mathbf{x}), \mathbf{x})$, $\forall \mathbf{x} \in \{\mathbf{x} = (x_1, x_2, x_3) : x_1 < x_2 < x_3$, $x_3 = (5, 0)$, x_1, and x_2 online connecting access point and $x_3\}$. Local maxima occur at $x_1 = x_2 = 1.9$ m and $x_1 = 1.5$ m, $x_2 = 3$ m.***

$\nu(\boldsymbol{\alpha}(\mathbf{x}), \mathbf{x})$ as depicted in Fig. 9(c) shows little change, and x_1 and x_2 cease progress. Fortunately, perturbations in the simulation are enough to pull the system out of the local minima so that it can achieve a global minimum at $t = 140$ s. As the number of agents increases, the frequency of local minima in $-\nu(\boldsymbol{\alpha}(\mathbf{x}), \mathbf{x})$ becomes more and more of an issue for local control. The addition of obstacles into the environment adds further difficulties as it not only affects feasible configurations $\mathbf{x}$ due to collision constraints, but also introduces nonsmooth components in the underlying point-to-point communication links $R_{ij}(\mathbf{x})$.

C. Global Planning

Gradient-based control will drive the system towards local minima of (13). However, as we consider larger teams with more complicated network topologies and complex environments with obstacles, local minima become more of an issue. We propose that a global search of (6) is necessary in order to accomplish the high-level situational-awareness tasks we are interested in.

To consider global search of (6), we redefine the problem to be more amenable to motion planning approaches from the robotics literature. Let X be a bounded connected open subset of $\mathbb{R}^{2N}$ that represents the full joint state space for the team of robots where $\mathbf{x}_0$ is the initial configuration of the team. In general, the goal region will be defined as $X_g = \{\mathbf{x} : \Psi(\mathbf{x}) < \Psi_{\min} + \delta\}$. In the telepresence application where a lead agent must visit the location $x_{\ell,g}$, $\Psi(\mathbf{x}) = \|x_\ell - x_{\ell,g}\|^2$ and $X_g = \{\mathbf{x} : \|x_\ell - x_{\ell,g}\| < \delta\}$. The obstacle region X_{obs} contains any configuration that places an individual robot on a physical obstacle and the infeasible

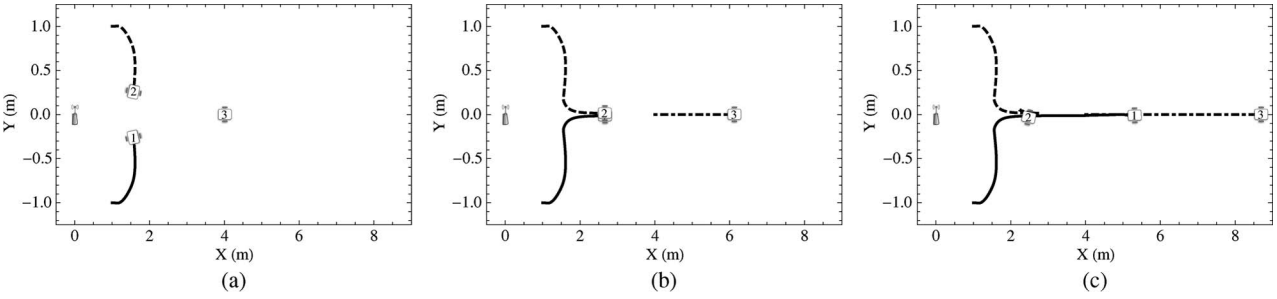

Fig. 7. ***Snapshots from a four robot trial. The end-to-end rates and probability of meeting the problem specifications are depicted in Fig. 9. $\Psi(\mathbf{x}(t)) = \|x_3 - (9, 0)\|$ for $t > 50$ s and $\Psi(\mathbf{x}(t)) = 0$ for $t \leq 50$ s: (a) $t = 40$ s; (b) $t = 75$ s; (c) $t = 140$ s.***

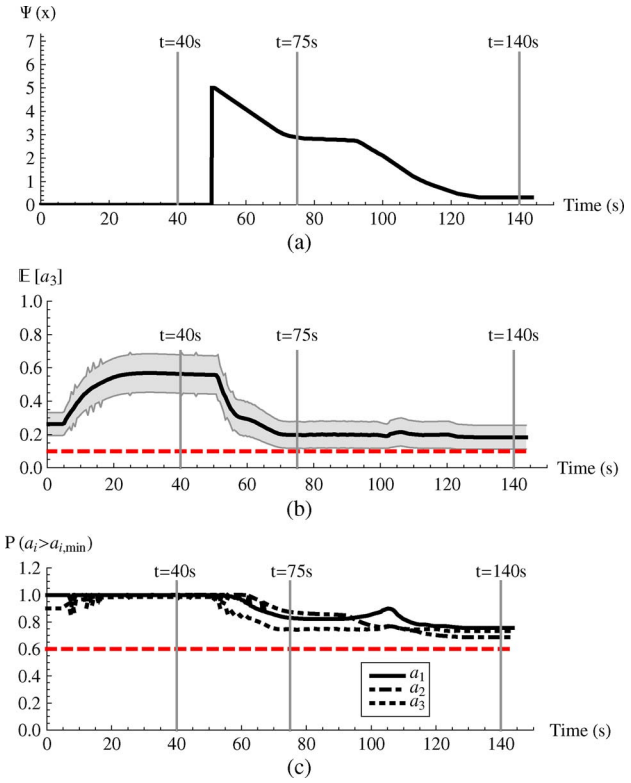

Fig. 9. *(a) Convergence of $\Psi(\mathbf{x})$, (b) end-to-end rates, and (c) probability of success for the four robot trial depicted in Fig. 7.*

region represents configurations where it is infeasible to satisfy the network constraint (11)

$$X_{\text{inf}} = \left\{ \mathbf{x} : \frac{\bar{a}_i^k(\boldsymbol{\alpha}, \mathbf{x}) - a_{i,\min}^k}{\sqrt{\bar{a}_i^k(\boldsymbol{\alpha}, \mathbf{x})}} < \Phi^{-1}(\epsilon) \right. \\ \left. \forall\, \boldsymbol{\alpha} \in \left\{ \alpha_{ij}^k : 0 \le \alpha_{ij}^k \le 1, \sum_{j,k} \alpha_{ij}^k \le 1 \right\} \right\}. \quad (16)$$

The free space X_{free} is then $X \setminus (X_{\text{obs}} \cup X_{\text{inf}})$. Finally, a path in X is parameterized by a scalar $s \ge 0$ and given by $\boldsymbol{\sigma} : [0, s] \to X$. A feasible path, and solution to our global-planning problem, is then $\boldsymbol{\sigma} : [0, s] \to X_{\text{free}}$ such that $\boldsymbol{\sigma}(0) = \mathbf{x}_{\text{init}}$ and $\boldsymbol{\sigma}(s) \in X_g$.

The dimensionality of our problem and the high computational cost of verifying a state as in X_{free} makes deterministic search algorithms impractical. Instead we turn to probabilistic search methods that offer good space filling properties and efficient exploration of an unknown space like the rapidly exploring random tree (RRT) algorithms [26]. The basic structure of an RRT, as depicted in Fig. 10, is to start with an initial point $\mathbf{x}_0$ and expand to fully explore the workspace, adding states in a tree structure $\mathcal{T}$ until a point $\mathbf{x} \in X_g$ is added to the tree $\mathcal{T}$. At each step of the RRT algorithm, we pick a random state $\hat{\mathbf{x}} = \text{RANDOMSTATE}(X, \mathcal{T})$ and select the point $\mathbf{x}_{\min} = \text{NEAREST}(\mathcal{T}, \hat{\mathbf{x}})$ that is closest to $\hat{\mathbf{x}}$ among those that have already been added to $\mathcal{T}$. We then attempt a virtual drive from $\mathbf{x}_{\min}$ to $\hat{\mathbf{x}}$ using the subroutine $\mathbf{x} = \text{EXTEND}(\mathbf{x}_{\min}, \hat{\mathbf{x}})$. The point $\mathbf{x}$ is the first intersection of this virtual path with the border of the free space region X_{free}, or, if the border is not reached, the random point $\hat{\mathbf{x}}$. The point $\mathbf{x}$ is then added to the tree $\mathcal{T}$ as a branch connected to the point $\mathbf{x}_{\min}$ that was closest to $\hat{\mathbf{x}}$ among the preexisting elements of the tree. The algorithm terminates when a point $\mathbf{x} \in X_g$ in the goal configuration is added to the tree.

A common problem encountered when applying RRT algorithms to high-dimensional state spaces is that computation of NEAREST is inefficient for increasing tree sizes. We adopt the strategy of storing the tree $\mathcal{T}$ in a KD-tree data structure which stores states in $\mathbb{R}^d$ by recursively subdividing based on alternating axis-aligned hyperplanes [27]. This enables approximate nearest neighbor calculations that maintain performance even as the dimension increases. However, there are two additional difficulties that arise when applying standard RRT algorithms to solve the specific high-dimensional network connectivity problem in (6): 1) the verification of feasible states as EXTEND is used to expand the tree towards $\hat{\mathbf{x}}$; and 2) the prohibitive cost of uniformly exploring X_{free} for our high-dimensional problem with slow-to-compute constraints. We discuss this two issues in the following two sections.

1) Efficient Verification of Feasible States: The $\text{EXTEND}(\mathbf{x}_1, \mathbf{x}_2)$ algorithm attempts to virtually drive the system from $\mathbf{x}_1$ towards $\mathbf{x}_2$ by successively verifying that points along the line connecting $\mathbf{x}_1$ and $\mathbf{x}_2$ are in X_{free}. It returns the state $\mathbf{x}$ as the closest state to $\mathbf{x}_2$ such that all states sampled with precision $\Delta\mathbf{x}$ between $\mathbf{x}_1$ and $\mathbf{x}_2$ are in X_{free}. In traditional motion planning applications,

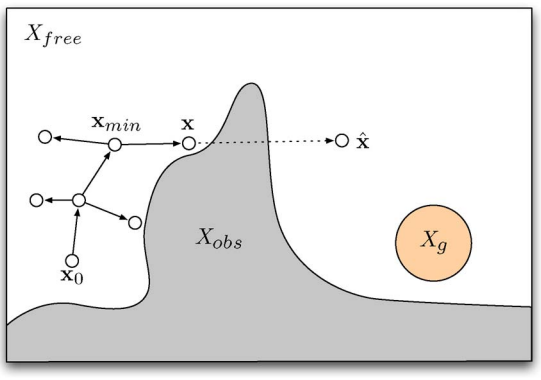

Fig. 10. *Graphical depiction of the RRT search process visualized in $\mathbb{R}^2$.*

verification that $\mathbf{x} \in X_{\text{free}}$ is based on an algebraic constraint or collision query with a multitude of efficient methods for doing so [28]–[30]. While the necessary computation to determine $\mathbf{x} \notin X_{\text{obs}}$ is typically small, computation of $\mathbf{x} \notin X_{\text{inf}}$ requires a solution of the SOCP in Section IV and can be costly for high-dimensional systems.

Consequently, we store $\boldsymbol{\alpha}(\mathbf{x})$ for every node in $\mathcal{T}$ and recompute $\boldsymbol{\alpha}(\mathbf{x})$ only when necessary to extend new states. By relying on the fact that an optimal robust routing solution $\boldsymbol{\alpha}(\mathbf{x})$ will be feasible for neighboring states, it is often possible to extend $\mathbf{x}$ towards $\hat{\mathbf{x}}$ without the costly overhead of numerical optimization.

2) Biased Space Sampling: Random states $\hat{\mathbf{x}}$ are chosen to sample the space $X \subset \mathbb{R}^{2N}$ according to a probability distribution $p_{\mathbf{x}}(\mathbf{x})$ representing the belief about configuration $\mathbf{x}$ being part of a feasible path $\boldsymbol{\sigma}(s)$. If nothing is known about $\boldsymbol{\sigma}(s)$, we choose $p_{\mathbf{x}}(\mathbf{x})$ uniform in the space X. In general, at least the final configuration is known in that $\boldsymbol{\sigma}(s) \in X_g$. We can then bias the distribution by designing $p_{\mathbf{x}}(\mathbf{x})$ to choose configurations in X_g with higher probability. Goal biasing improves efficiency of RRT algorithms by reducing the number of samples necessary to find a feasible path $\boldsymbol{\sigma}(s)$ in the high-dimensional space $X \subset \mathbb{R}^{2N}$.

In many cases of interest, however, the volume of X_g is comparable to the volume of X and goal biasing offers little improvement over uniform sampling. In, for example, the telepresence application, the goal position of the leader $x_{\ell,g}$ is known, but the positions of the remaining robots are free. Thus, goal biasing would reduce the exploration cost along the components associated with x_ℓ but keep the cost of exploring the remaining $2(N-1)$ dimensions fixed. To further reduce exploration cost in this case we construct a prediction $\tilde{X}_g \subset X_g$ of the final configuration and bias sampling towards this prediction.

Constructing a final configuration prediction $\tilde{X}_g$ is task specific. We describe here a method applicable to the telepresence application. To determine the configuration prediction $\tilde{X}_g$ we determine configuration predictions $\tilde{X}_{i,g}$ for each robot and compute $\tilde{X}_g$ as the Cartesian product of these individual sets, i.e., $\tilde{X}_g = \prod_{i=1}^{N} \tilde{X}_{i,g}$. Notice that for the lead robot we can make $\tilde{X}_{\ell,g} = \{x_\ell \in \mathbb{R}^2 : \|x_\ell - x_{\ell,g}\| < \delta\}$.

Observe now that $X \subset \mathbb{R}^{2N}$ is the Cartesian product $X = \prod_{i=1}^{N} X_i$ of the N decoupled spaces $X_i \in \mathbb{R}^2$ corresponding to each individual robot. If we further assume a homogeneous team of robots, then all robots operate in the same space $X_i = Y$, with a common set of physical obstacles Y_{obs}, and consequently a common free space $Y_{\text{free}} = Y \setminus Y_{\text{obs}}$. It follows that the joint free space X_{free} is also a Cartesian product of N identical sets Y_{free} minus those configurations for which a network cannot be established with sufficient reliability

$$X_{\text{free}} = (Y_{\text{free}})^N \setminus X_{\text{inf}}. \tag{17}$$

While infeasible network configurations are captured by X_{inf} as given in (16), X_{free} can otherwise be described by the free space of individual robots.

To exploit this observation, we first determine an obstacle-free path $\gamma : [0, s] \to \mathbb{R}^2$ such that $\gamma(0) = x_0$ is the position of the operating center and $\gamma(s) \in X_{\ell,g}$. The obstacle-free path $\gamma : [0, s] \to Y_{\text{free}}$ is split into $N-1$ equal length segments γ_k. The ith robot is then assigned to a segment by the function $k(i)$ based on Euclidian distance to its midpoint. Segments are enlarged to define the region $\tilde{X}_{i,g}$ for $i \neq 0, \ell$. Since this is a heuristic for the goal configuration, the only requirement on $\tilde{X}_{i,g}$ is that $\gamma_{k(i)} : [0, s] \to \tilde{X}_{i,g}$. A typical choice is

$$\tilde{X}_{i,g} = \left\{ x_i : \min_s \left\| x_i - \gamma_{k(i)}(s) \right\| < \tilde{d}_g \right\}$$

where $\tilde{d}_g$ is a parameter controlling the enlarged size of $\tilde{X}_{i,g}$. The predicted final configuration is then computed as the Cartesian product $\tilde{X}_g = \prod_{i=1}^{N} \tilde{X}_{i,g}$. The construction of $\tilde{X}_g$ described above is based on the heuristic that a feasible goal configuration in an environment with obstacles will *resemble* a line-of-sight communication chain. Increasing the size of $\tilde{X}_g$ with large values of $\tilde{d}_g$ limits the implication of this assumption.

D. Experimental Results

The randomized motion planner is able to find feasible configurations that allow target servicing at

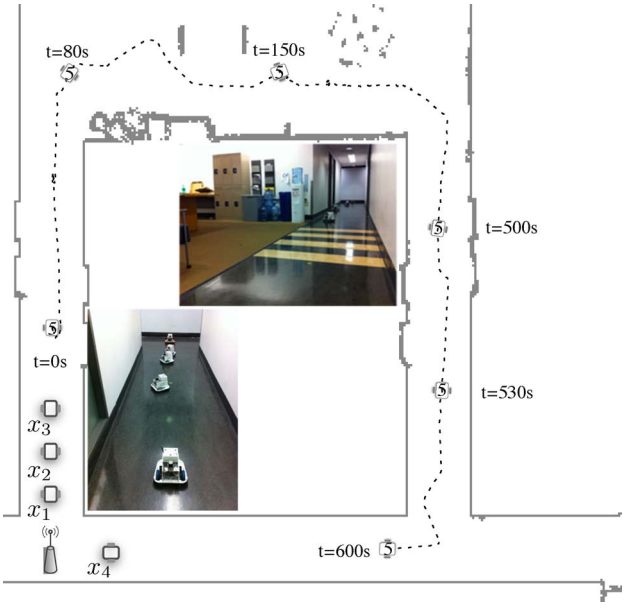

Fig. 11. *The global planning task for five robots in a complex environment. We depict the series of waypoints that x_5 must visit and the initial conditions for x_1, x_2, x_3, x_4. The inset images depict our experimental system deployed in the environment.*

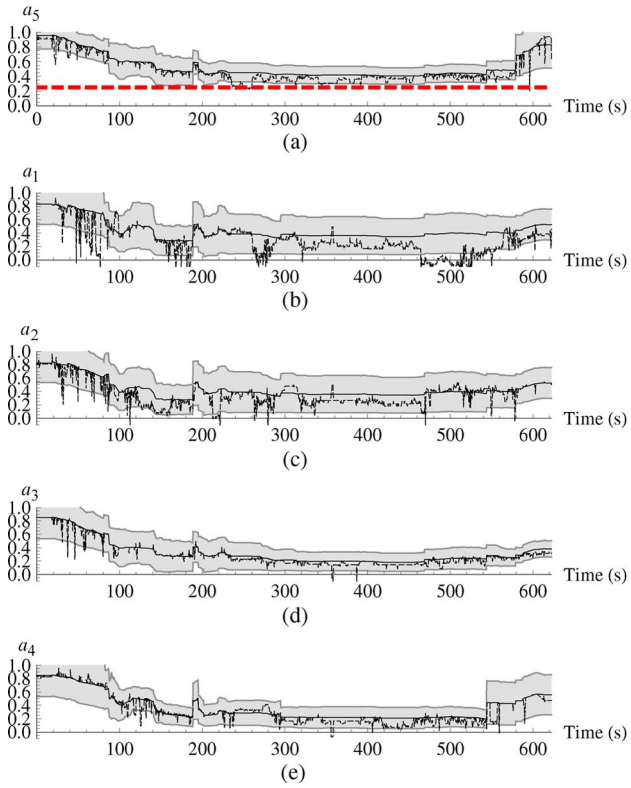

Fig. 12. *The end-to-end rates of the nodes during the global planning experiment depicted in Fig. 11. In each plot, the shaded envelope depicts $\bar{a}_i$ and the variations that occur with probability $\epsilon = 0.75$ based on $\tilde{a}_i$ while the dashed black line depicts the instantaneous rate $\hat{a}_i$. Note that $a_{5,\min} = 0.25$ while all other $a_{i,\min} = 0$.*

positions not attainable with the local control approach from Section V-A. It can additionally provide a feasible sequence of configurations to get to the target configuration. To test the global planner on a system with one fixed access point and five robots, we introduce a sequence of task potential functions $\Psi_1(\mathbf{x}), \Psi_2(\mathbf{x}), \ldots, \Psi_M(\mathbf{x})$ that require the *lead* node x_5 to visit a sequence of positions while the remaining four robots act as relays to support end-to-end communication with the access point of $a_{5,\min} = 0.25$. We require that this end-to-end rate be satisfied with probability $\epsilon = 0.75$. The problem is made more complicated by the introduction of obstacles that not only block robot motions but also degrade received signal strength in (7) by 7.6 dBm when line-of-sight is lost.

The global planning algorithm described in Section V-C is queried to find a feasible path $\sigma_i : [0, s] \rightarrow \mathbb{R}^{10}$ for each task $\Psi_i(\mathbf{x})$ in order such that $\sigma_i(0) = \sigma_{i-1}(s)$ and $\sigma_1(0) = \mathbf{x}_{\text{init}}$. The trajectory of the lead node x_5 for the concatenation of paths $\sigma = \sigma_1 | \ldots | \sigma_M$ is depicted in Fig. 11. After solving for a feasible path, $\dot{\mathbf{x}}^{\text{des}}(t)$ is computed so that $\mathbf{x}(t)$ follows σ.

We conduct an experimental deployment of five robots controlling the feasible trajectory σ for the problem depicted in Fig. 11. In this experiment, each robot is equipped with a 2.4-GHz Zigbee radio that is used to broadcast and receive packets in order to measure instantaneous received signal strength that can be used to estimate the supported communication rate $\hat{R}_{ij}(t)$ between node i and j at time t. Using these measurements, in conjunction with the network routing solution $\boldsymbol{\alpha}(t)$, we can estimate the actual supported end-to-end rate at time t for each node i, $\hat{a}_i(\boldsymbol{\alpha}(t), \mathbf{x}(t))$. The metric for the performance of our approach is then that each $\hat{a}_i(\boldsymbol{\alpha}(t), \mathbf{x}(t)) > a_{i,\min}$ with probability ϵ across the duration of the trial.

Fig. 12 depicts the predicted $\bar{a}_i(\boldsymbol{\alpha}(t), \mathbf{x}(t))$ and measured $\hat{a}_i(\boldsymbol{\alpha}(t), \mathbf{x}(t))$ end-to-end rate of each node for the duration of the experiment. First, note that over the entire trajectory produced with our global planning methods, the prediction of end-to-end rates always satisfies the problem specification that $\mathbf{P}[a_i(\boldsymbol{\alpha}, \mathbf{x}) \geq a_{i,\min}] \geq \epsilon$. Second, when we examine the actual end-to-end rates based on real measurements of point-to-point wireless channels we see that it is generally true that $\hat{a}_i(\boldsymbol{\alpha}(t), \mathbf{x}(t)) \geq a_{i,\min}$. In fact, if we analyze this constraint across the duration of the experiment, we find that the fraction of time spent below

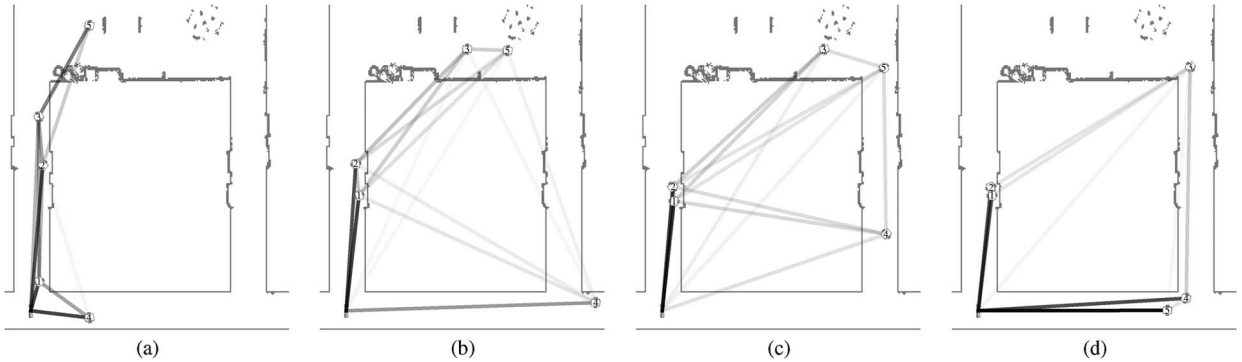

Fig. 13. *Snapshots of the network configuration during the experimental deployment based on global planning to satisfy the task in Fig. 11. The end-to-end rate performance is depicted in Fig. 12. (a) $t = 100$ s; (b) $t = 278$ s; (c) $t = 482$ s; (d) $t = 622$ s.*

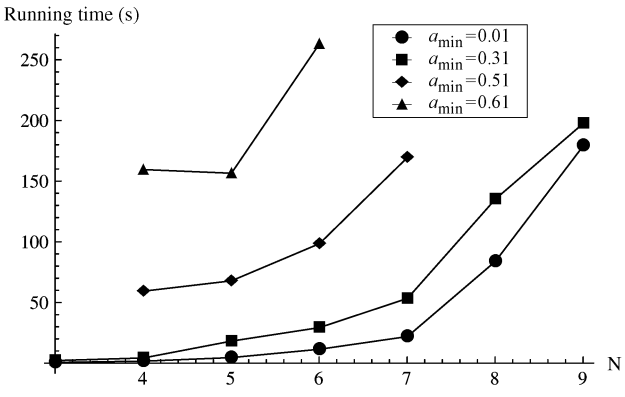

Fig. 14. ***Running time for a benchmark planning task computed with an increasing number of robots N and different end-to-end rate thresholds a_{min}. Since the global planning algorithm is randomized, each point is the average running time over ten trials.***

the minimum threshold for each of the instantaneous rates $\hat{a}_1, \hat{a}_2, \hat{a}_3, \hat{a}_4,$ and $\hat{a}_5$ is 9.2%, 0.8%, 0.3%, 0.6%, and 2.9%, respectively.

Representative configurations from the experiment are depicted in Fig. 13. In Fig. 13, at $t = 100$ s, the predicted goal state $\tilde{X}_g$ assumes the shortest line-of-sight path which is the left hallway, i.e., a similar result to the reactive methods in our local control algorithm. As the system transitions to Fig. 13(b), where the lead node x_5 has been tasked to a waypoint in the right hallway, the prediction for $\tilde{X}_g$ shifts to a chain of relays going through the right hallway. This shift in the basic topology of $\tilde{X}_g$ focuses on exploration of the joint state space so that x_4 moves towards a configuration that will *lower* the performance of the network over the short term. As node x_5 completes the desired loop, it utilizes x_4 as a relay channel and is able to maintain the desired end-to-end rate. It is this dramatic shift in network topology that highlights the advantage of our global planning approach as we are able to accomplish continuous end-to-end rate maintenance that would not be possible with a purely reactive method.

With regards to the running time of the randomized global planning algorithm, we note that it is difficult to characterize the performance of randomized search algorithms. One factor is the complexity of X_{free} which is determined both by physical obstacles in the environment as well as the constraints placed on feasible network configurations, i.e., the solution to $\boldsymbol{\alpha}(\mathbf{x})$. Fig. 14 depicts the running time for a benchmark environment as we increase the dimension of the problem N by adding robots to the system. We also vary the end-to-end rate threshold a_{min} that must be maintained for the lead node to demonstrate how more strict network requirements also increase complexity of the search space.

VI. CONCLUSION

We propose a system architecture that provides end-to-end connectivity for autonomous teams of robots as they pursue operator-assigned tasks. This architecture is composed of a cyber component that determines the configuration of the wireless network and a physical component that handles mobility. Because the performance of point-to-point wireless links is difficult to predict, we adopt a stochastic model for supported rates and develop optimal robust solutions to the wireless routing problem. The strengths of the proposed approach are that: 1) network availability takes into account specific thresholds on the end-to-end rates of individual nodes; 2) considering the stochastic nature of communication channels network availability is further defined in probabilistic terms; 3) by specifically searching for robust solutions, we obtain configurations with spatial diversity and increased probability of success in the face of difficult to predict communication channels.

Robust routing solutions are integrated with a physical component that handles mobility either through local control or global planning. Local controllers converge to solutions that are local minima of the concurrent mobility and network routing problem. The global planner performs a randomized biased search in the space of spatial configurations in an attempt to jointly solve the mobility and optimal network routing problems. We perform experiments to demonstrate that the global planning algorithm succeeds in navigating a complex environment while ensuring that end-to-end communication rates meet or exceed prescribed values within a target failure tolerance. Global planners are able to discover dramatic shifts in communication topology that allow for continuous communication that is not possible with purely local methods. ■

REFERENCES

[1] J. Fink, "Communication for teams of networked robots," Ph.D. dissertation, Elect. Syst. Eng., Univ. Pennsylvania, Philadelphia, PA, 2011.

[2] S. Thrun, W. Burgard, and D. Fox, Probabilistic Robotics. Cambridge, MA: MIT Press, 2005.

[3] M. C. DeGennaro and A. Jadbabaie, "Decentralized control of connectivity for multi-agent systems," in *Proc. 45th IEEE Conf. Decision Control*, San Diego, CA, Dec. 2006, pp. 3628–3633.

[4] M. Ji and M. Egerstedt, "Distributed coordination control of multiagent systems while preserving connectedness," *IEEE Trans. Robot.*, vol. 23, no. 4, pp. 693–703, Aug. 2007.

[5] M. Zavlanos and G. Pappas, "Potential fields for maintaining connectivity of mobile networks," *IEEE Trans. Robot.*, vol. 23, no. 4, pp. 812–816, Aug. 2007.

[6] E. Stump, A. Jadbabaie, and V. Kumar, "Connectivity management in mobile robot teams," in *Proc. IEEE Int. Conf. Robot. Autom.*, Pasadena, CA, May 2008, pp. 1525–1530.

[7] M. Schuresko and J. Cortes, "Distributed motion constraints for algebraic connectivity of robotic networks," *J. Intell. Robot. Syst.*, vol. 56, no. 1–2, pp. 99–126, Sep. 2009.

[8] D. P. Spanos and R. M. Murray, "Motion planning with wireless network constraints," in *Proc. Amer. Control Conf.*, Portland, OR, Jun. 2005, pp. 87–92.

[9] G. Notarstefano, K. Savla, F. Bullo, and A. Jadbabaie, "Maintaining limited-range connectivity among second-order agents," in *Proc. Amer. Control Conf.*, Minneapolis, MN, Jun. 2006, pp. 2124–2129.

[10] E. Stump, "Control for localization and visibility maintenance of an independent agent using robotic teams," Ph.D. dissertation, Mech. Eng. Appl. Mech., Univ. Pennsylvania, Philadelphia, PA, 2009.

[11] C. Perkins, E. Belding-Royer, and S. Das, "Rfc3561: Ad hoc on-demand distance vector (aodv) routing," Internet RFCs, 2003.

[12] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile Comput.*, no. 353, pp. 153–181, 1996.

[13] H. Lundgren, E. Nordstrom, and C. Tschudin, "The gray zone problem in ieee 802.11b based ad hoc networks," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 6, no. 3, pp. 104–105, Jul. 2002.

[14] A. Ribeiro, Z.-Q. Luo, N. D. Sidiropoulos, and G. B. Giannakis, "Modelling and optimization of stochastic routing for wireless multihop networks," in *Proc. 26th Annu. Joint Conf. IEEE Comput. Commun. Soc.*, Anchorage, AK, May 2007, pp. 1748–1756.

[15] D. DeCouto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multihop wireless routing," in *Proc. Int. ACM Conf. Mobile Comput. Netw.*, San Diego, CA, Sep. 2006, pp. 134–146.

[16] I. Stojmenovic, A. Nayak, and J. Kuruvila, "Design guidelines for routing protocols in ad hoc and sensor networks with a realistic physical layer," *IEEE Commun. Mag.*, vol. 43, no. 3, pp. 101–106, Mar. 2005.

[17] A. Hsieh, A. Cowley, V. Kumar, and C. J. Taylor, "Towards the deployment of a mobile robot network with end-to-end performance guarantees," in *Proc. IEEE Int. Conf. Robot. Autom.*, Orlando, FL, May 16–18, 2006, pp. 2085–2090.

[18] Y. Mostofi, "Communication-aware motion planning in fading environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 3169–3174.

[19] M. Zavlanos, A. Ribeiro, and G. Pappas, "Mobility and routing control in networks of robots," in *Proc. Conf. Decision Control*, Atlanta, GA, Dec. 15–17, 2010, pp. 7545–7550.

[20] Y. Mostofi, A. Gonzalez-Ruiz, A. Gaffarkhah, and L. Ding, "Characterization and modeling of wireless channels for networked robotic and control systems—A comprehensive overview," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2009, pp. 4849–4854.

[21] J. Fink, N. Michael, A. Kushleyev, and V. Kumar, "Experimental characterization of radio signal propagation in indoor environments with application to estimation and control," in *Proc. Int. Conf. Intell. Robot. Syst.*, St. Louis, MO, Oct. 2009, pp. 2834–2839.

[22] J. Fink and V. Kumar, "Online methods for radio signal mapping with mobile robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, Anchorage, AK, May 2010, pp. 1940–1945.

[23] R. Shorey, A. Ananda, M. C. Chan, and W. T. Ooi, *Mobile, Wireless and Sensor Networks: Technology, Applications and Future Directions*. New York: Wiley, 2006.

[24] L. R. Klein, *A Textbook of Ecometrics*. Evanston, IL: Row, Peterson, 1953.

[25] Y. Wu, A. Ribeiro, and G. B. Giannakis, "Robust routing in wireless multi-hop networks," in *Proc. Conf. Inf. Sci. Syst.*, Mar. 14–16, 2007, pp. 637–642.

[26] J. Kuffner, Jr. and S. LaValle, "RRT-connect: An efficient approach to single-query path planning," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2000, vol. 2, pp. 995–1001.

[27] A. Atramentov and S. M. LaValle, "Efficient nearest neighbor searching for motion planning," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, pp. 632–637.

[28] M. C. Lin, D. Manocha, J. Cohen, and S. Gottschalk, "Collision detection: Algorithms and applications," in *Algorithms for Robotic Motion and Manipulation*, J.-P. Laumond and M. H. Overmars, Eds. Wellesley, MA: A.K. Peters, 1997, pp. 129–142.

[29] B. Mirtich, "V-Clip: Fast and robust polyhedral collision detection," Mitsubishi Electron. Res. Lab., Tech. Rep. TR97-05, 1997.

[30] M. C. Lin and J. F. Canny, "Efficient algorithms for incremental distance computation," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1991, pp. 1008–1014.

ABOUT THE AUTHORS

Jonathan Fink (Student Member, IEEE) received the B.S. degree in electrical and computer systems engineering from Rensselaer Polytechnic Institute, Troy, NY, 2004 and the Ph.D. degree in electrical and systems engineering from the University of Pennsylvania, Philadelphia, in 2011.

He is a Research Scientist with the U.S. Army Research Laboratory, Adelphi, MD. His research interests include collaboration and planning for multirobot systems with emphasis on communication modeling and planning for network connectivity.

Dr. Fink has received the best student paper award at the 2009 Robotics: Science and Systems Conference (RSS 2009).

Alejandro Ribeiro (Member, IEEE) received the B.Sc. degree in electrical engineering from the Universidad de la Republica Oriental del Uruguay, Montevideo, Uruguay, in 1998 and the M.Sc. and Ph.D. degrees in electrical engineering from the Department of Electrical and Computer Engineering, the University of Minnesota, Minneapolis, in 2005 and 2007, respectively.

Since 2008, he has been an Assistant Professor at the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia. 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 wireless communications and networking, distributed signal processing, and wireless sensor networks.

Dr. Ribeiro is a Fulbright scholar and the recipient of a National Science Foundation CAREER award in 2010 and best student paper awards at the 2005 and 2006 International Conference on Acoustics, Speech, and Signal Processing (ICASSP).

Vijay Kumar (Fellow, IEEE) received the Ph.D. degree in mechanical engineering from The Ohio State University, Columbus, in 1987.

He is the UPS Foundation Professor and the Deputy Dean for Education in the School of Engineering and Applied Science, University of Pennsylvania, Philadelphia. He has been on the Faculty in the Department of Mechanical Engineering and Applied Mechanics with a secondary appointment in the Department of Computer and Information Science at the University of Pennsylvania since 1987.

Dr. Kumar is a Fellow of the American Society of Mechanical Engineers (ASME). He has served on the editorial boards of the IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, *Journal of Franklin Institute*, the IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, *ASME Journal of Mechanical Design*, *the ASME Journal of Mechanisms and Robotics*, and the *Springer Tracts in Advanced Robotics (STAR)*. He is the recipient of the 1991 National Science Foundation Presidential Young Investigator award, the Lindback Award for Distinguished Teaching, and the 1997 Freudenstein Award for significant accomplishments in mechanisms and robotics. He has won best paper awards at the Distributed Autonomous Robotic Systems 2002, the International Conference on Robotics and Automation 2004, the International Conference on Robotics and Automation 2008, the Robotics: Science and Systems 2009, and the Distributed Autonomous Robotic Systems 2010. He is a Distinguished Lecturer in the IEEE Robotics and Automation Society and an elected member of the Robotics and Automation Society Administrative Committee.