# Adaptive Communication in Multi-robot Systems Using Directionality of Signal Strength

**Stephanie Gil, Swarun Kumar, Dina Katabi and Daniela Rus**

**Abstract** We consider the problem of satisfying communication demands in a multi-agent system where several robots cooperate on a task and a fixed subset of the agents act as mobile routers. Our goal is to position the team of robotic routers to provide communication coverage to the remaining client robots. We allow for dynamic environments and variable client demands, thus necessitating an adaptive solution. We present an innovative method that calculates a mapping between a robot's current position and the signal strength that it receives along *each spatial direction*, for its wireless links to every other robot. We show that this information can be used to design a simple positional controller that retains a quadratic structure, while capturing the behavior of wireless signals in real-world environments. Notably, our approach does not necessitate stochastic sampling along directions that are counter-productive to the overall coordination goal, nor does it require exact client positions, or a known map of the environment.

## 1 Introduction

Multi-agent robotic systems perform many complex tasks through coordination, such as cooperative search of an environment, consensus, rendezvous, and formation control [1–3]. As cooperation is at the core of multi-robot tasks, the performance of these systems directly hinges on the robots' ability to communicate reliably. To maintain certain communication guarantees, these systems need a mapping of communication

S. Gil (✉) · S. Kumar · D. Katabi · D. Rus
Massachusetts Institute of Technology, Boston, MA, USA
e-mail: sgil@mit.edu

S. Kumar
e-mail: swarun@cmu.edu

D. Katabi
e-mail: dk@mit.edu

D. Rus
e-mail: rus@mit.edu



M. Inaba and P. Corke (eds.), *Robotics Research*, Springer Tracts in Advanced Robotics 114, DOI 10.1007/978-3-319-28872-7_4

quality to robot placement. Producing such a mapping however is quite challenging [4]. Past literature employs two broad strategies to address this challenge: On the one hand, there is the Euclidean disk model which assumes that the signal quality of a link is a function of distance between the communicating vehicles. This model is deterministic and simple, and hence when incorporated in a robotic controller, yields simple positional optimizations for a wide range of collaborative tasks [1–3]. Unfortunately, the Euclidean model is too simplistic and fails to represent wireless signals in realistic environments [4]. On the other hand, there are stochastic sampling methods [4–6] that measure the wireless signal strength in a robot's vicinity to fit parameters for intricate probabilistic communication models. While such methods are not oblivious to wireless channels, they require exploratory sampling [7] along directions that may be counter-productive to the overall coordination goal. Further, they often assume the knowledge of parameters based on the structure and material composition of the environment.

Our objective here is to (i) present a novel method for capturing the spatial variation of wireless signals in the local environment *without* sampling along counter-productive directions, or requiring information about the environment and/or the channel's distributions and (ii) derive a control formulation that maintains the structural (quadratic) simplicity allowed by the Euclidean disk model while accounting for this wireless channel feedback. First, we introduce an innovative approach for mapping communication quality to robot placement. We calculate a mapping between a robot's current position and the signal strength that it receives along *each spatial direction*, for every wireless link with other robots. This is in contrast to existing methods [5, 6], which compute an aggregate signal power at each position but cannot distinguish the amount of signal power received from each spatial direction.

Second, we construct an optimization for positioning a team of robot routers to provide communication coverage to an independent set of client vehicles using the directional information provided by our mapping. We aim for a solution that is adaptive to variable communication quality demands by the clients, as well as changes in the wireless channels due to natural fluctuations or a dynamic environment. Being able to measure the profile of signal strength across spatial directions in real-time yields a much more capable controller. For example, the controller uses the profile to find directions of movement that yields better communication quality. The profile also helps estimate the confidence with which the controller can improve signal power by navigating the robot along any of these directions. The confidence can then be used to control the speed of the robot, thereby improving stability and convergence time. Furthermore, the controller can leverage the entire profile of signal strength across directions, to optimize communication with multiple robots by choosing a direction of movement corresponding to a strong signal that strikes trade-offs between competing demands. Interestingly, we show that such optimizations can be formulated in terms of simple quadratic costs, similar in spirit to the disk model. Further, they can be made independent of environment-dependent parameters, or even client positions.

A key question remains: how do we calculate the signal strength along each spatial direction? The naive approach would use directional antennas, a type of antenna that

receives signals only from a cone in space. Unfortunately, directional antennas are bulky and have low spatial resolution [8] (about 60°), making them ill-suited for small agile robots. To address this problem, we employ Synthetic Aperture Radar (SAR), a technique that leverages movement to emulate a high-resolution directional antenna [9]. In order to achieve this, we must derive a method for implementing SAR using off-the-shelf wireless cards, a challenging task since these devices are not intended for this purpose.

We implement our method in a multi-robot testbed that has two robotic routers serving three robotic clients. We conduct our experiments in different indoor environments without providing the robotic controller the environment map or the clients' positions. We observe the following: (1) Our system consistently positions the robotic routers to satisfy the robotic client demands, while adapting to changes in the environment and fluctuations in the wireless channels; (2) Compared to the disk model [1, 2] and the stochastic approach [10, 11] under identical settings, our system converges to accurately satisfy the communication demands, unlike the disk model, while significantly out-performing the stochastic method in terms of empirical convergence rate (see Fig. 8 in Sect. 5.4).

**Contributions**: The contributions of this paper are three-fold: (1) We present a method to enable a robotic receiver to find the profile of signal strength across spatial directions for each sender of interest. To this end, we perform synthetic aperture radar (SAR) techniques using standard Wi-Fi packets exchanged between two independent nodes; (2) We develop an optimization that leverages this directional signal profile to position robotic routers to satisfy heterogeneous communication demands of a network of robotic clients, while adapting to real-time environmental changes; (3) We implement our design and demonstrate its empirical gains in comparison to both the disk model and the stochastic method.

## 2 Related Work

Our work is related to past papers on multi-robot coordination to achieve a collaborative task while supporting specific communication demands [4–6, 10, 12]. These papers recognize the importance of measuring the signal strength on real-world wireless links to model communication quality. Such papers typically build analytical models of the signal strength on a wireless link to account for the effects of distance, obstacles, and reflections on the signal. The models are then supplemented with measurement data. While these approaches provide a more realistic integration of robot coordination with communication constraints than the disk model, they often necessitate parameter fitting that are environment-dependent. Further, they require sampling of signal strength along stochastic directions that may be counter-productive to the overall coordination goal. In comparison to these papers, we introduce a system that captures the different *directions* of a signal, as opposed to only its magnitude at a particular position. This allows us to satisfy variable demands from multiple robotic

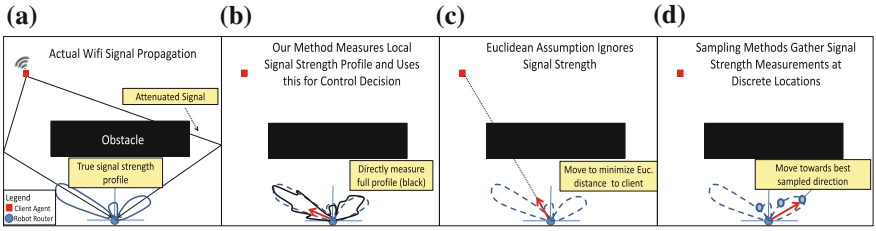


**Fig. 1** Schematic drawings demonstrating the differences between the current method and previous methods

clients in an environment-oblivious fashion, and without sampling the signal along stochastic directions (Fig. 1).

Our work is also related to past work on synthetic aperture radar (SAR). SAR allows us to exploit the natural movements of robots to calculate the signal strength along each spatial direction. Past work on SAR however assumes a single device that transmits a signal and receives its reflections [9, 13–15], and none of this work can use off-the-shelf Wi-Fi cards. In contrast, we present a system that extracts SAR information from standard Wi-Fi packets transmitted between different devices.

## 3 Problem Statement

We consider a mobile network with two classes of members, $n$ robotic clients (or *clients*) whose positions are not controlled, and a team of $k$ robotic routers whose mobility we control. Our goal is to position the robotic routers to provide adaptive wireless communication coverage to the clients, while allowing variable communication quality demands for all clients, and where exact client positions are unknown. For each client $j \in [n] = \{1, \ldots, n\}$, we define demanded communication quality $q_j > 0$,[1] and achieved communication quality $\rho_{ij}$ to each router $i$ (where $i \in [k]$), both expressed in terms of *Effective Signal to Noise Ratio* (ESNR) that has a direct mapping to rate in Mb/s [16].[2] Additionally, let every client $j$ be given an importance $\alpha_j > 0$. We define the notion of *service discrepancy* for each pair of robots $(i, j)$ to be the difference between the demanded and achieved communication quality scaled by the importance of the client.

$$w_{ij} = \max(\alpha_j(q_j - \rho_{ij})/q_j, 0) \tag{1}$$

[1] Note that all quantities in this section are time-dependent; we omit this dependency for simplicity.

[2] We choose to work with ESNR values rather than rates since the rates supported on a link are discretized (non-continuous).

Physically, this is the fraction of the client's communication demand that remains to be satisfied, scaled by $\alpha_j$. Denote by $c_i \in \mathbb{R}^d$ the position of the $i$th robot router and by $p_j \in \mathbb{R}^d$ the position of the $j$th client[3] and $C_t = \{c_{1,t}, \ldots, c_{k,t}\}$ is the set of all router positions at time $t$. Given a cost $g$ in terms of signal quality, communication demands, and agent positions, we wish to position each robotic router to minimize the largest discrepancy of service between routers and clients. However, the true form of this function $g$ has an intricate dependence on the position of the client, router, and the environment. Thus an inherent challenge to solving this problem is approximating the influence of spatial positioning on communication quality in a way general to different environments. We have a joint goal to (1) find $f_{ij} : [-\frac{\pi}{2}, \frac{\pi}{2}] \to \mathbb{R}$ (a relation capturing directional information about the signal quality between $i$ and $j$), and an approximation $\tilde{g}$ of $g$ that is a cost characterizing the anticipated communication quality for the router-client pair $(i, j)$ at a proposed router position $c_i$, and (2) use this cost to optimize router positions to minimize the service discrepancy to each client. Formally,

**Problem 1** Find a mapping

$$f_{ij} : \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \to \mathbb{R} \tag{2}$$

that maps spatial direction to wireless signal strength directly from channel measurements, and a cost

$$\tilde{g}(c_i, C_t, w_{ij}, f_{ij}) > 0 \tag{3}$$

that is independent of the environment and client positions, has a simple quadratic form, and whose minimization over $c_i$ directly relates to increasing signal quality. We aim to find robot router positions, $C_{t+1}$ that minimize the maximum service discrepancy over all clients $j$ by solving the following min–max problem:

$$C_{t+1} = \arg\min_{c_i \in C} \{\max_j \min_i \tilde{g}(c_i, C_t, w_{ij}, f_{ij})\} \tag{4}$$

Intuitively, the solution to this optimization problem favors "fair" solutions where the maximum service discrepancy is minimized over all clients. We dedicate the next sections of this paper to (1) Developing a method that computes $f_{ij}$ as the profile of signal qualities along each direction $\theta$ for each link $(i, j)$ found directly from channel measurements; and (2) Developing an optimization framework that utilizes this directional information to handle trade-offs between competing client demands, and position all routers to jointly minimize the maximum service discrepancy over the links in the communication network.

[3] In this paper we mainly consider $d = 2$ although all concepts are extensible to $d = 3$.

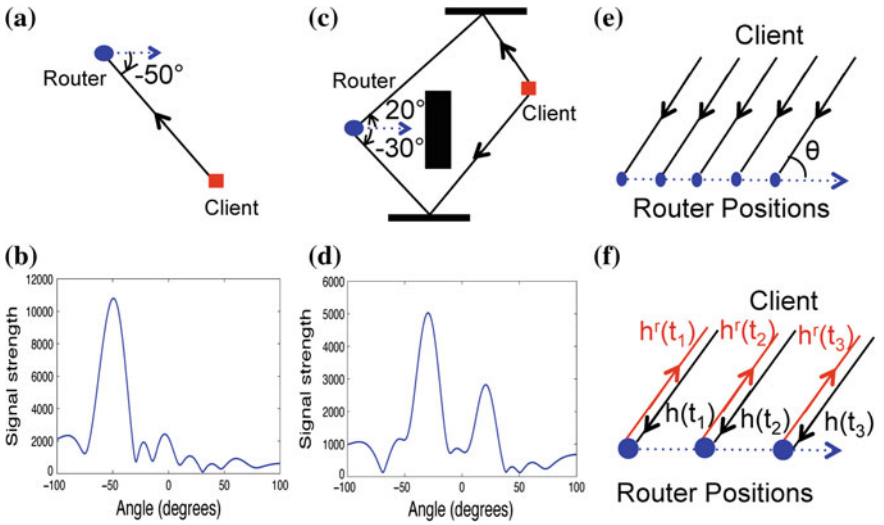


**Fig. 2** **a**, **c** LOS and NLOS topologies annotated with signal paths. **b**, **d** $f(\theta)$ of the signal in LOS and NLOS. **e** Shows how $\theta$ is defined in SAR. **f** Shows $h(t_i)$, the forward channel from transmitter to receiver and $h^r(t_i)$, the reverse channel from receiver to transmitter at time $t_i$

# 4 Approach

## *4.1 Computing the Directional Power Profile of a Wireless Link*

In this section, we develop the first component of the solution of Problem 1; namely, we derive a method to calculate $f(\theta)$, the mapping which captures the strength of the signal from a robotic client to its router along each direction $\theta$.[4] Where this mapping can be updated often, roughly once every 6 cm of motion.

Before we explain how we compute $f(\theta)$, we describe this function to help understand the information it captures. Assume we have a robotic client and router, where the router moves along some trajectory. We will define the direction $\theta$ relative to the tangent to the router's trajectory at each point. Consider the scenario in Fig. 2a, where the robotic client is in line-of-sight at $-50°$ relative to the robotic router, which is moving along the horizontal axis. In this case, one would expect $f(\theta)$ to have a single dominant peak at $-50°$, as shown in Fig. 2b. Now consider the more complex scenario in Fig. 2c, where the environment has some obstacles and one of these obstacles obstructs the line-of-sight path between the router and its client. In

[4] For simplicity, we denote $f_{ij}(\theta)$ as $f(\theta)$ as we consider only the single link between robotic router $i$ and client $j$ for the rest of this section.

this case, $f(\theta)$ would show two dominant peaks at 20° and −30° that correspond to the two reflected paths from surrounding obstacles, as shown in Fig. 2d.

**Advantage over Sampling Methods**: One may estimate $f(\theta)$ by sampling the signal power similar to stochastic techniques [5, 10, 11]. In this case, one has to move the router along each direction, compute the power in all these new positions relative to the first, and draw the profile $f(\theta)$. Unfortunately, this approach leads to much wasted exploration. This is because the signal power does not change reliably when the robot moves. For example, if the robot moves for 5 or 10 cm, it is very likely that the resulting change in the signal power is below the variability in noise. Hence, measurements of power over short distances are likely to be marred by noise. To obtain reliable measurements of changes in the signal power, the robot has to move significantly along potentially counter-productive paths.

To address this limitation, our approach relies on the channel phase as opposed to the power. Specifically, at any position the wireless channel can be expressed as a complex number $h(t)$ [17]. The magnitude of this complex channel captures the signal power (more accurately, its square-root). The phase of the channel has traditionally been ignored by robotic systems. However, the phase changes rapidly with motion. For Wi-Fi signals at a frequency of 5 GHz, the phase of the channel rotates by $\pi$ every 3 cm. This far exceeds any rotation due to noise variability. Thus, by measuring channels as complex numbers and tracking changes in its phase as the robot moves, we reliably estimate signal variation without much exploration. In the next section, we explain how to use a technique called synthetic aperture radar (SAR) to extract the received signal strength along each direction from changes in channel phase. Note that SAR does not need exploring all directions; the robot can move along its path without extra exploration or sampling. SAR uses the resulting variations in channel phase over distances of a few centimeters to find $f(\theta)$.

### 4.1.1 Synthetic Aperture Radar (SAR)

Synthetic Aperture Radar (SAR) enables a single antenna mounted on a mobile device to estimate the strength of the signal received along every spatial direction. We leverage the natural motion of a robotic router to implement SAR and measure $f(\theta)$ for each of its robotic clients using an omni-directional antenna. To do so, the robotic router measures the channel $h(t)$ from its client as it moves along any straight line. The straight line path over which the router acquires data is on the order of half a wavelength (centimeters); assuming the source is stationary and the router either moves at a known constant velocity or its position is known for the traversal time window, then a sufficient amount of usable channel data can be collected. This means every few centimeters the router can have an updated measurement of $f(\theta)$, for all values of $\theta$.

Specifically, Let $h(t)$ for $t \in \{t_0, \ldots, t_m\}$ be the $m+1$ most recent channel measurements, corresponding to the robot moving a distance $d(t_0) \ldots d(t_m)$. SAR computes

the received signal strength across spatial directions $f(\theta)$ as:

$$f(\theta) = \left| \sum_t h(t) e^{-j\frac{2\pi}{\lambda} d(t) cos\theta} \right|^2, \quad (5)$$

where $\lambda$ is the wavelength of the Wi-Fi signal. We refer the reader to [18] for the analysis of this standard SAR equation. At a high level, the multiplying terms $e^{-j\frac{2\pi}{\lambda} d(t) cos\theta}$ in Eq. (5) project the channels $h(t)$ along the direction of interest $\theta$ by compensating for incremental phase rotations introduced by the robot's movement to all paths of the signal arriving along $\theta$.

Note that SAR finds the signal power from every angle $\theta$ simply by measuring the channels,[5] without needing prior tuning to any given direction. In fact, moving by around a wavelength (about 6 cm) is sufficient to measure the full profile of $f(\theta)$.

Therefore, SAR is a natural choice for autonomous robotic networks since it exploits the mobility of the robots to compute $f(\theta)$. Further, it only requires the robot to move along a small straight line along any arbitrary direction, and does not require it to explore directions counter-productive to the overall coordination goal. Note that SAR requires only the relative position of the robotic router $d(t)$ and the both the magnitude and phase of the channel $h(t)$. It does not require the topology of the environment nor the exact location of the transmitter.

### 4.1.2 Challenges in Implementing SAR on Independent Wireless Devices

A key challenge in adapting SAR to multi-robot systems is that all past SAR-based solutions [9, 13, 15] are for radar-like applications, where a single device transmits a radar signal and receives its reflections off an imaged object, e.g., an airplane. However, in our scenario the transmitter and receiver are completely independent wireless devices (i.e., the robotic client and router, respectively). This means that the transmitter robot and the receiver robot have different frequency oscillators. In practice, there is always a small difference between the frequency of two independent oscillators. Unfortunately, even a small offset $\Delta_f$ in the frequency of the oscillators introduces a time varying phase to the wireless channel.

For instance, let $h(t_0)$, $h(t_1)$, ..., $h(t_m)$ be the actual wireless channel from the robotic client to the robotic router at times $t_0, t_1, \ldots, t_m$. The channel observed by the router from its client $\hat{h}(t_0)$, $\hat{h}(t_1)$, ..., $\hat{h}(t_m)$ are given by:

$$\hat{h}(t_0) = h(t_0), \quad \hat{h}(t_1) = h(t_1) e^{-2\pi \Delta_f (t_1 - t_0)}, \ldots, \quad \hat{h}(t_m) = h(t_m) e^{-2\pi \Delta_f (t_m - t_0)}. \quad (6)$$

Hence, the phase of the channels are corrupted by time-varying values due to the frequency offset between the transmitter and the receiver. Fortunately, we can correct

[5] Of course, the resolution at which $\theta$ is available depends on the number of channel measurements.

for this offset using the well-known concept of channel reciprocity [17]. Specifically, let $h^r(t)$ denote the reverse channel from the robotic router to its client, as shown in Fig. 2f. Reciprocity states that the ratio of the forward and reverse channels stays constant over time, subject to frequency offset, i.e. $h^r(t) = \gamma h(t)$, where $\gamma$ is constant. Further, the frequency offset in the reverse direction $\Delta_f^r$ is negative of the offset in the forward direction, i.e. $\Delta_f^r = -\Delta_f$. Thus, the observed reverse channels $\hat{h}^r(t_0)$, $\hat{h}^r(t_1), \ldots, \hat{h}^r(t_m)$ are given by:

$$\hat{h}^r(t_0) = h^r(t_0), \quad \hat{h}^r(t_1) = h^r(t_1)e^{2\pi\Delta_f(t_1-t_0)}, \quad \ldots, \quad \hat{h}^r(t_m) = h^r(t_m)e^{2\pi\Delta_f(t_m-t_0)}. \quad (7)$$

Multiplying Eqs. 6 and 7 above and using $h^r(t) = \gamma h(t)$, we have $\hat{h}(t)\hat{h}^r(t) = h(t)h^r(t) = \gamma h(t)^2 \Rightarrow h(t) = \sqrt{\hat{h}(t)\hat{h}^r(t)/\gamma}$. Hence we re-write Eq. (5) as:

$$f(\theta) = \left| \sum_t \sqrt{\hat{h}(t)\hat{h}^r(t)} e^{-j\frac{2\pi}{\lambda}d(t)cos\theta} \right|^2, \quad (8)$$

where the constant scaling $\gamma$ is dropped for simplicity. Hence, to measure $f(\theta)$ the router and client simply need to measure their channels at both ends.[6] In the next section, we explain how we leverage $f(\theta)$ on each link to control the position of multiple robotic routers to meet the clients' communication demands.

## 4.2 Optimizing Robotic Router Placement Using Channel Feedback

In this section, we target the problem of placing a team of mobile router vehicles at locations such that they provide wireless coverage to client vehicles, each with different communication demands. Specifically, using as input the channel feedback $f_{ij}(\theta)$ derived in the previous section, we aim to find a function $\tilde{g}$ that can be optimized over router positions such that:

$$C_{t+1} = \arg\min_C \{\max_j \min_{c_i \in C} \tilde{g}(c_i, C_t, w_{ij}, f_{ij})\} \quad (9)$$

Our focus in this section is to find a function $\tilde{g}$ that has three desirable properties: (1) It is quadratic; (2) It allows for trade-offs between clients with competing demands as captured by the service discrepancies $w_{ij}$; and (3) It is independent of client positions $p_j$. In the rest of this section, we show how to capitalize the rich spatial information

[6] In practice, the router and client transmit back-to-back packets with a small gap $\delta \approx 200\,\mu s$ to obtain $\hat{h}^r(t+\delta)$ and $\hat{h}(t)$, respectively. The router collects these values and approximates $\hat{h}(t)\hat{h}^r(t)$ as $\hat{h}(t)\hat{h}^r(t+\delta)e^{-j2\Delta_f\delta}$. The router computes this 10 times per second (an overhead of just 0.1 %).

provided by $f_{ij}(\theta)$, to derive a cost $\tilde{g}$ possessing the three desired qualities. We can then optimize this cost to complete our objective of robot router placement that best satisfies the communication demands of the clients.

### 4.2.1 A Generalized Distance Metric for Incorporation of Channel Feedback

Our first goal is to translate signal quality over all directions, $f_{ij}(\theta)$, to a cost $\tilde{g}$ that can be optimized over router positions. We begin with the case where all positions are known and extend to the position independent case in Sect. 4.2.3. Consider a single router-client pair $(i, j)$ located at positions $(c_i, p_j)$. A disk model approach to service this client does not use $f_{ij}(\theta)$ at all. Instead, it relates improving communication quality between the router and client to reducing the Euclidean distance between them, i.e. the cost $\tilde{g} := dist(p_j, c_i)$. The appeal of such a cost is in its simple quadratic form that can be easily optimized. Unfortunately, the cost is oblivious to the actual wireless channel at the client and fails to capture the current service discrepancy which can be large even at small distances (say, due to obstacles).

Our system avoids this pitfall, while retaining simplicity, by incorporating real-time channel feedback into a generalized distance metric. In particular, we do not assume that the shortest distance for enabling better communication between two robots is the straight line path between them, but rather the path along the $\theta_{\max}$, the direction of maximum signal strength from the mapping $f_{ij}(\theta)$. Thus, the client is recommended to move towards $\mathbf{v}_{\theta_{\max}}$, the unit vector along $\theta_{\max}$.

Importantly, the recommended heading direction $\mathbf{v}_{\theta_{\max}}$ may exhibit variation due to noise or multipath affecting the wireless link. To account for these effects, while not over-fitting to noise, we leverage the entire $f_{ij}$ signal profile to design a *confidence* metric $\sigma_{ij}$ in heading direction. Intuitively, $\sigma_{ij}$ captures the"variance" of $f_{ij}$ around $\theta_{\max}$.[7] We would like to encode this quantity into our controller such that $\mathbf{v}_{\theta_{\max}}$ directions of high confidence are followed more aggressively (larger displacements along these directions), and the opposite is true of $\mathbf{v}_{\theta_{\max}}$ directions with low confidence. Specifically, $\sigma_{ij}$ falls under the following categories: (1) $\sigma_{ij} < 1$: Indicates a high confidence in $\mathbf{v}_{\theta_{\max}}$ due to a sharp peak in $f_{ij}$. The robot is moved at higher speeds; (2) $\sigma_{ij} \approx 1$: Indicates that $f_{ij}$ is noisy, so the robot must move slowly; (3) $\sigma_{ij} > 1$: Indicates that $f_{ij}$ has multiple significant peaks owing to multi-path. We study this case, and particularly the opportunity it presents for making trade-offs between clients, more elaborately in Sect. 4.2.2.

We can use the heading direction and confidence to design a cost function $\tilde{g}$ that accurately captures the cost of communication in the spatial domain. Interestingly, we can express this cost as a generalized distance metric called the *Mahalanobis distance*. The square of the Mahalanobis distance is a cost function (paraboloid) with ellipsoidal level sets (Fig. 3). We design our cost by orienting these level sets so that the direction of steepest descent is along $\mathbf{v}_{\theta_{\max}}$. We then skew the ellipsoidal

[7] Mathematically, $\sigma_{ij} = \sum_{\theta} [(\theta - \theta_{\max})^2 f_{ij}(\theta)] / \sum_{\theta} [(\theta - \theta_{\max})^2 \text{mean}\{f_{ij}(\theta)\}]$.

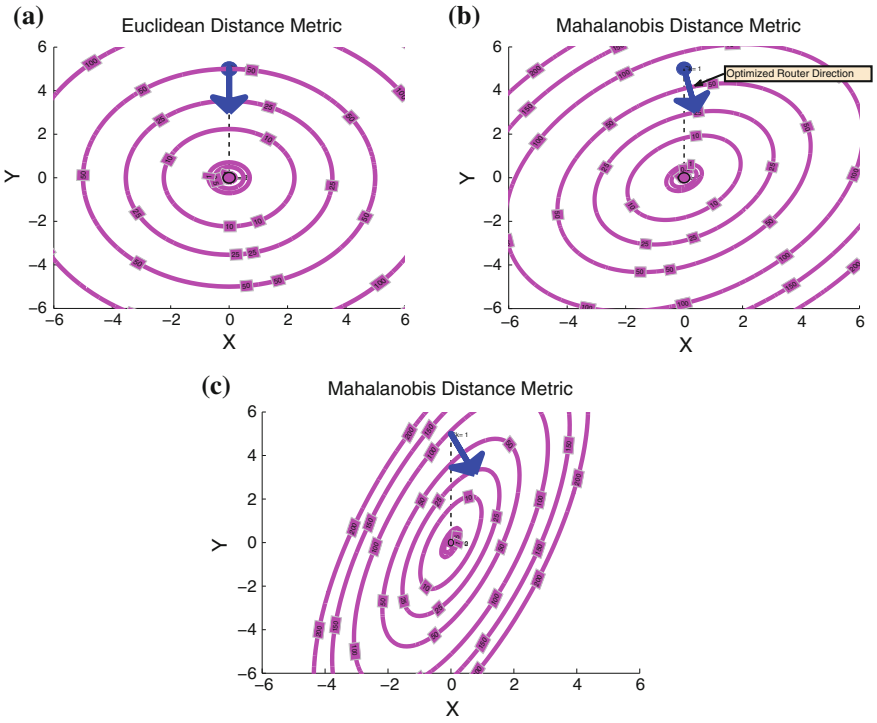


**Fig. 3** These plots show the level sets of Euclidean and Mahalanobis distance functions. **a** Euclidean distance. **b** Mahalanobis (Low Conf). **c** Mahalanobis (High Conf)

level sets using the confidence $\sigma_{ij}$, so that a higher confidence translates to a steeper descent. Mathematically, the Mahalanobis distance is given by:

**Definition 1** (*Mahalanobis Distance*) Given a positive definite matrix $M \in \mathbb{R}^{dxd}$, a vector $x \in \mathbb{R}^d$, and a vector $y \in \mathbb{R}^d$, the Mahalanobis Distance between $x$ and $y$ is:

$$\text{dist}_M(x, y) = \sqrt{(x - y)^T M (x - y)} \tag{10}$$

Euclidean distance is a special case of the Mahalanobis distance (see Fig. 3a) with $M = I$ where $I$ is the identity matrix of appropriate dimension.

Here, $M = Q\Lambda Q^T$ is a positive-definite matrix, where $Q$ consists of orthogonal eigen-vectors and $\Lambda$ contains the corresponding eigen-values. We simply set one of the eigen-vectors of $Q$ to the heading direction $\mathbf{v}_{\theta_{\max}}$. To skew the ellipsoid, we set the ratio of the eigen values $\{\lambda_1, \lambda_2\}$ in $\Lambda$ to the confidence $\sigma^2$, i.e. $\lambda_2/\lambda_1 = \sigma^2$, where $\lambda_1$ is the eigen-value corresponding to $\mathbf{v}_{\theta_{\max}}$. For e.g., In Fig. 3b, where $\sigma \approx 1$ (i.e. poor confidence), the level sets are nearly circular, leading to a shallow descent in cost; while Fig. 3c, where $\sigma < 1$ (i.e. high confidence), the level sets are skewed, leading

to a steep descent in cost along $\mathbf{v}_{\theta_{\max}}$. In other words, the cost function has an elegant geometric interpretation, akin to Euclidean distance, but is derived directly from channel measurements. Further, the cost function $\tilde{g} := \text{dist}_{M_{ij}}(p_i, c_j)$ from Eq. (10) is quadratic, a desirable property for optimizations.

### 4.2.2 Network Trade-Offs

In this section, we show how our optimization framework readily extends to a multi-agent scenario and study the different trade-offs. We show that via the setting of two parameters, both set automatically from wireless channel data, the resulting positional controller can be made to greedily optimize one client vs. strike trade-offs between multiple clients. First, we focus on managing service discrepancies specified by $w_{ij}$. $w_{ij}$ aims to bias the controller by assigning higher weight to users with larger service discrepancies. To do this, we scale the cost function $\tilde{g} = \text{dist}_{M_{ij}}(p_i, c_j)$ by the square of the discrepancy $w_{ij}^2$ to optimize:

$$r_M(P, C) = \max_{p_j \in P} \min_{c_i \in C} \{w_{ij}^2 \text{dist}_{M_{ij}}(p_i, c_j)\} \tag{11}$$

Second, we highlight the subtle role played by the confidence $\sigma_{ij}$ in managing network trade-offs. For instance, consider a scenario with two clients: 1 and 2, where client-1 demands greater communication quality (as specified by $w_{ij}$'s). Suppose client-1 has a highly confident $\mathbf{v}_{\theta_{\max}}$ as shown in Fig. 4a (i.e. $\sigma_{ij} < 1$). As expected, the robotic router is directed towards client-1 as shown in Fig. 4c. In the more interesting scenario in Fig. 4b, client-1's confidence is poor due to multiple peaks in the signal profile $f_{ij}$ (i.e. $\sigma_{ij} > 1$). Here, the router strikes a trade-off and services client-2 instead, as this may potentially benefit client-1 as well due to the multipath recognized in client-1's $f_{ij}(\theta)$ map. The intuition behind this is simple. Eq. (13) above, scales the ellipsoidal cost function based on the discrepancies $w_{ij}$'s. However, recall that the ellipsoidal cost function is steep (or shallow) depending on whether the confidence is high (or low) and this is attained by setting the ratio of eigenvalues $\lambda_2/\lambda_1$ of $M_{ij}$. In extremely low confidence scenarios such as Fig. 4b, the higher value of discrepancy of client-1 is masked by its low value of confidence. This balances the trade-off in favor of client-2, despite a lower discrepancy.

### 4.2.3 A Position-Independent Solution

A simple relaxation to the cost from the previous section frees the optimization of using client positions, while maintaining its simple structure and desirable properties developed above. Consider a given stepsize $\gamma > 0$. We replace client positions $p_j$ in

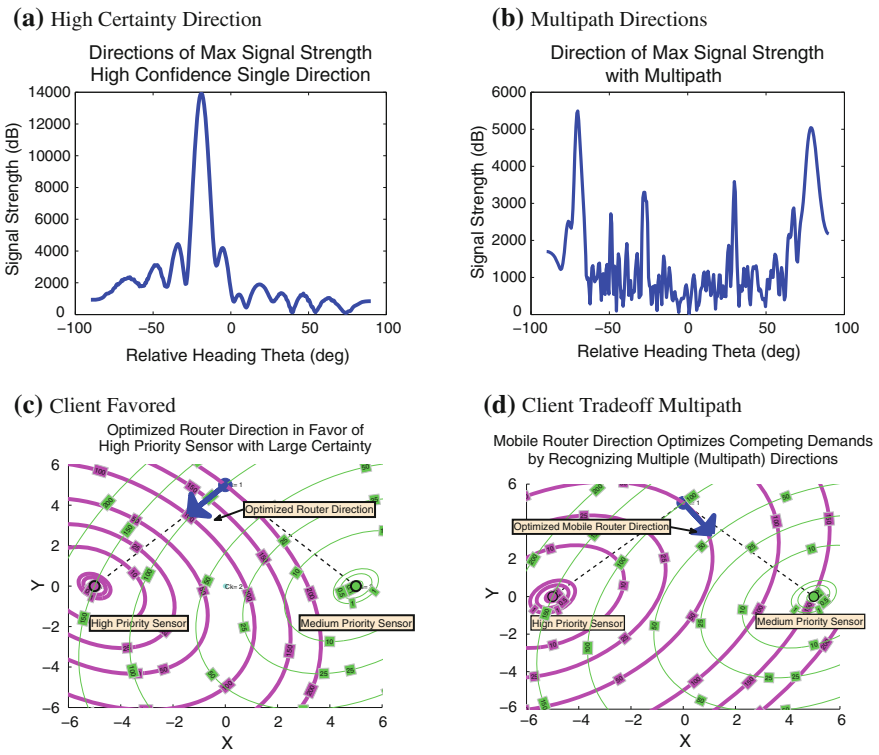


**Fig. 4** **Trade-offs between Clients**: **a**, **b** show the $f_{ij}(\theta)$ map for the high demand client; **c**, **d** show the optimized router direction

Eq. (13) with "virtual" positions $p'_{ij}$:

$$p'_{ij} = c_{i,t} + \gamma w_{ij} \mathbf{v}_{\theta_{\max}} \tag{12}$$

Loosely, a client is no longer directly observed but rather estimated to be along the relative direction $\mathbf{v}_{\theta_{\max}}$ and at a distance of $\gamma w_{ij}$ with respect to the $i$th router. As before, $\mathbf{v}_{\theta_{\max}}$, is the heading direction associated with the maximum stength signal direction $\theta_{\max}$. As a client's demand is better satisfied by router $i$, the service discrepancy $w_{ij}$ tends to 0 and the client is perceived as being closer to router $i$. The intuition here is that routers better equipped to service a particular client as reflected by the $w_{ij}$ term, will view the client as "closer" and those routers with a weaker signal to the same client will view this client as farther away. This results in a natural method of assigning client nodes to routers by effectively sensing over the wireless channels. Our final cost takes the form:

$$r_M(C) = \max_{j \in \{1,\ldots,n\}} \min_{c'_i \in C'} \{\text{dist}^2_{M_{ij}}(c_{i,t} + \gamma w_{ij} \mathbf{v}_{\theta_{\max}}, c'_i)\} \tag{13}$$

By expanding the squared per-link cost $\text{dist}^2_{M_{ij}}(c_i + \gamma w_{ij}\mathbf{v}_{\theta_{\max}}, c'_i)$ from Eq. (13):

$$(c'_i - c_{i,t})^T M_{ij}(c'_i - c_{i,t}) - 2\gamma w_{ij}\lambda_{\theta_{ij}}\mathbf{v}^T_{\theta_{\max}}(c'_i - c_{i,t}) + \gamma^2 w^2_{ij}\lambda_{\theta ij} \tag{14}$$

we note that as $w_{ij} \rightarrow 0$ the first term in Eq. (14) favors stable solutions where $c'_i = c_{i,t}$, i.e. the router reaches a static solution when all of its assigned clients have zero service discrepancy.

Finally for a set of routers with positions $C$, $r_M(C)$ reflects the cost of the client with the largest service discrepancy. The positions $C$ that minimize $r_M(C)$ can be found by solving this optimization as a second order cone program as in [19, 20].

As defined in our problem statement, Problem 1, we have found a set of quadratic costs $g(p_j, c_i, C_t, w_{ij}, f_{ij}) = (p'_{ij}(c_{i,t}, w_{ij}, \mathbf{v}_{\theta_{\max}}) - c_i)^T M_{ij}(p'_{ij}(c_{i,t}, w_{ij}, \mathbf{v}_{\theta_{\max}}) - c_i)$ that can be optimized in the desired min–max formulation from (4) in order to find an optimized robotic router placement for our wireless network.

## 5 Experimental Results

We evaluated our system on a five-node testbed with two routers and three clients. Each node was an ASUS 1015PX netbook equipped with an Intel 5300 Wi-Fi card mounted on an iRobot Create robot. We implemented SAR by modifying the iwlwifi driver on Ubuntu 10.04. We used the 802.11 CSI tool [21] to obtain channel information ($\hat{h}(t)$ in Eq. (8)). The routers communicated with a central laptop emulating the base for control information and human input. We performed our experiments in a room with a Vicon motion capture system to aid robot navigation. Our testbed contains obstacles to simulate both line-of-sight and non-line-of-sight scenarios.

### *5.1 Computing Direction of Maximum Signal Strength*

We first observe how effectively our system computes the direction of maximum signal strength $\theta_{\max}$, on a wireless link. We consider a single client, serviced by a robot router that is: (1) In direct line-of-sight (LOS) as shown in Fig. 5a. (2) In possible non-line-of sight (NLOS) scenarios due to obstacles as shown in Fig. 5b. We drive the robot router in a lawn-mover pattern and get $\theta_{\max}$ at regular intervals.

**Results**: Figure 5a and b depict the gradient field with the arrows indicating $\theta_{\max}$ in LOS and NLOS, respectively. The gradient field in LOS accurately directs the robot router towards the client regardless of its initial position. In NLOS, the robot is directed away from obstacles so that controller can route around obstacles to improve signal strength. We stress that $\theta_{\max}$ is found locally at the router purely via wireless channels and its own position, *without* prior knowledge of the environment. Further, the plots are not static and naturally *change over time*, especially in dynamic settings. Thus our system obtains instantaneous $\theta_{\max}$ values locally in real-time.

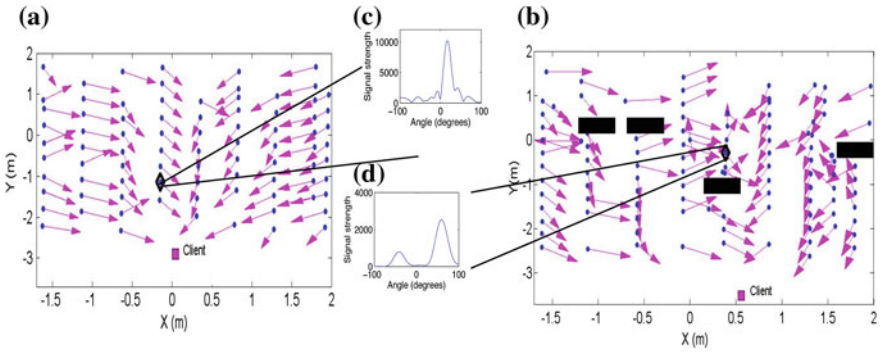


**Fig. 5** Gradient field of $\theta_{\max}$ and power profile for **a** Line-of-sight and **b** Non-line-of-sight

Figure 5c and d plot $f_{ij}(\theta)$, the power profile of the signal along different directions, for a candidate location in line-of-sight and non-line-of-sight scenarios, respectively. Clearly, the power profile in line-of-sight is dominated by a single peak at $\theta_{\max}$, directed along the line-of-sight path to the client. In contrast, the power profile in non-line-of-sight close to an obstacle has two significant peaks, each corresponding to reflected paths along walls or other objects in the environment.

## 5.2 *Controlling Router Trajectory to Satisfy Client Demands*

We evaluate how a single robotic router finds a trajectory to satisfy the demands of three clients (specified in terms of effective signal-to-noise ratio or ESNR) using $\theta_{\max}$ on each link. We consider the candidate non-line-of-sight setting in Fig. 6a. The router is unaware of exact client positions or the layout of the environment.

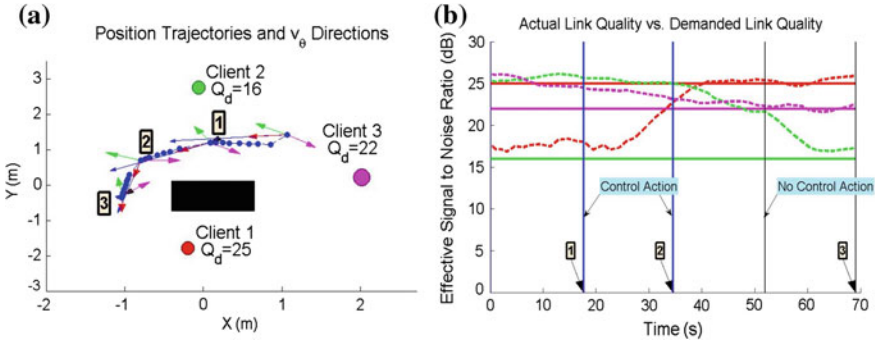


**Fig. 6** **a** Depicts testbed with robot router servicing three clients in a candidate non-line-of-sight setting. The *blue line* depicts the trajectory, and *colored arrows* indicate instantaneous $\theta_{\max}$ for the corresponding clients. **b** Plots the ESNR across time (as *dotted lines*) for each client through the experiment. *Solid lines* denote client demands

**Results**: Figure 6a depicts the trajectory of the robotic router in blue. The colored arrows denote the recommended $\mathbf{v}_{\theta_{\max}}$ directions for each client at every control point. The figure shows how the robot performs non-zero control actions until it eventually satisfies network demands. Figure 6b tracks the ESNR of the clients across time (dotted lines). The plot shows that the ESNR demands of each client (solid lines) are satisfied upon convergence. Note that the whenever the robot decides to follow the $\mathbf{v}_{\theta_{\max}}$ of a client at a control point (vertical line), the client's ESNR increases. This validates our claim that following a heading direction based on $\mathbf{v}_{\theta_{\max}}$ indeed improves the ESNR of the corresponding client.

## 5.3 *Aggregate System Results*

We evaluate our full system with two robot routers serving three clients with different ESNR demands. We perform the experiment in line-of-sight (LOS) and non-line-of-sight (NLOS) settings as shown in the inset maps of Fig. 7b and 7d respectively. We repeat the experiment five times in each setting and plot the results.

**Results**: Figure 7a and b plot the mean and variance of ESNR over time across experiments for each client (dotted colored lines) in LOS and NLOS. Clearly, each client's ESNR demand (solid lines) is satisfied at the converged position across

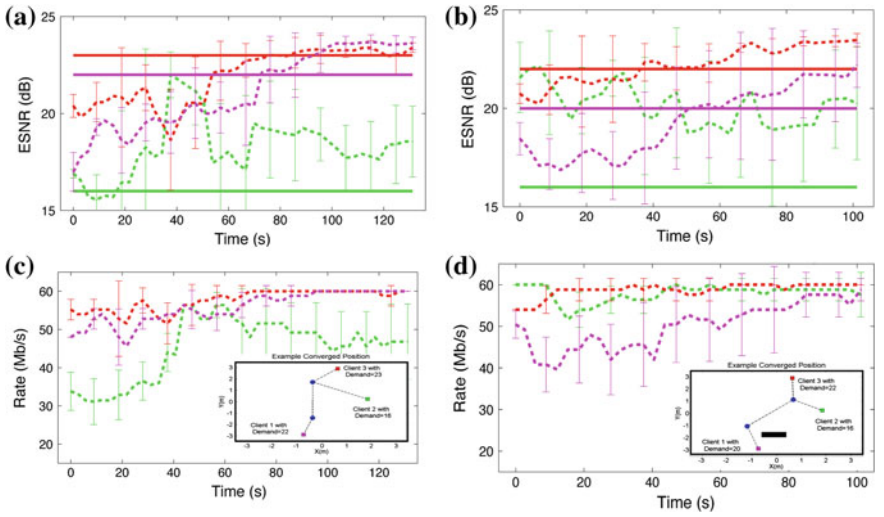


**Fig. 7** Agregate results obtained over 5 runs show demands are consistently met even in the presence of obstacles as demonstrated by the candidate converged solutions. **a** ESNR versus time (Line of sight). **b** ESNR versus time (Non line of sight). **c** Rate versus time (Line of sight). **d** Rate versus time (Non line of sight)

experiments. Figure 7c and d plot the corresponding aggregate link rate across time, which follows the same trend as the ESNR [16].[8] The inset plots in Fig. 7c and d depict the final converged position of the routers (blue dots) in LOS and NLOS. The results show that our system consistently satisfies client demands while adapting to real-time changes in wireless channels, even in the presence of obstacles.

### *5.4 Comparison with Existing Schemes*

We test our method against two other popular approaches to the communication problem in robotics: (1) Euclidean Disk Model as used in [1, 2], where communication constraints are in terms of Euclidean distance; (2) Stochastic Gradient Approach, where we implement the Simultaneous Perturbation method (SPSA) [11] for estimating the gradient of signal power by sampling the ESNR (which provides greater granularity than RSSI), along randomized directions, similar to the approach utilized by [10]. For the generation of each direction in the SPSA method we use a Bernoulli random variable (as in [11]) and diminishing step sizes satisfying the conditions stated in [11] for convergence. Our largest step size was allowed to be the same maximum vehicle velocity of $v_c$ for all experiments. We consider a robotic router and three clients, each with an ESNR demand of 20 dB. We repeat the experiment five times in the non-line-of-sight environment in Fig. 8b–d. In each instance, we measure $r_{\max}$, the maximum ratio of ESNR demand versus the ESNR achieved among all three clients. In particular, $r_{\max}$ is below one at the converged position (i.e. all client demands are satisfied), and above one otherwise.

**Results**: Figure 8a plots the aggregate mean and variance of $r_{\max}$ across time, for all the three approaches. Figure 8b–d shows a candidate trajectory adopted by the robotic router for the three schemes. The plots demonstrate while the disk model converges quickly to a solution, ignorance of the wireless channels leads to solutions not meeting client demands; especially in non-line-of-sight settings. In contrast, the stochastic gradient approach (in blue), which sample the instantaneous ESNR, eventually satisfies network demands. However, the convergence is often laborious as the router often traverses counter-productive directions (see Fig. 8c). Indeed such techniques are noisy at low signal power, as even a large change in distance translates to a small change in signal power (a well-studied problem in communications literature [22–24]). Figure 8c shows that this leads to areas at non-line-of-sight or far distances from the client, where the robot easily gets lost.

Our method leverages full information of the channel, including signal power and phase, to find the direction of signal power as opposed to its magnitude. The result is an algorithm that converges to positions that satisfy network demands while not necessitating counter-productive exploration steps of a pure sampling approach.

[8] Note that the data-rate is capped by 60 Mb/s causing the plot to appear flat at times unlike ESNR.

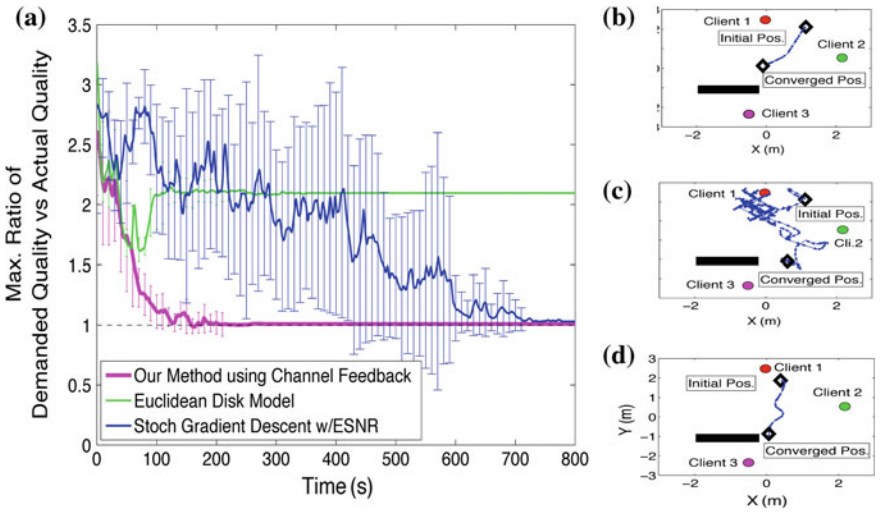


**Fig. 8** Plots comparing our method against the Euclidean disk model and a stochastic gradient descent method based on ESNR. Our method both converges to a position that meets communication demands, and converges quickly along an efficient path. **a** Comparison. **b** Euclidean disk model. **c** Stochastic method. **d** Our method

## 5.5 *Robustness to Dynamic Obstacle Positions*

We evaluate how our system adapts to changes in the environment without an *a priori* known map. Consider two robotic routers and three clients in an environment with an obstacle located initially as shown in Fig. 9a. We allow the robot routers to navigate to their converged positions. At $t = 120$ s, we move the obstacle to a different location as in Fig. 9c, and let the routers re-converge.

**Results**: Figure 9b and c depict the converged position of the routers before and after the obstacle was moved. Figure 9d plots the data-rate across time for each client. The plot shows that our system satisfies client demands at the initial position. Further it recovers from the sharp fall in data-rate to one of the clients to successfully re-converge after the obstacle is moved.

## 5.6 *Complex Indoor Environments*

We evaluate our system in a large complex indoor environment with concrete walls and columns. We place a robotic router and client and line-of-sight (LOS) and non-line-of-sight (NLOS) as in Fig. 10. We trace the router's gradient field towards the client starting from multiple initial positions.

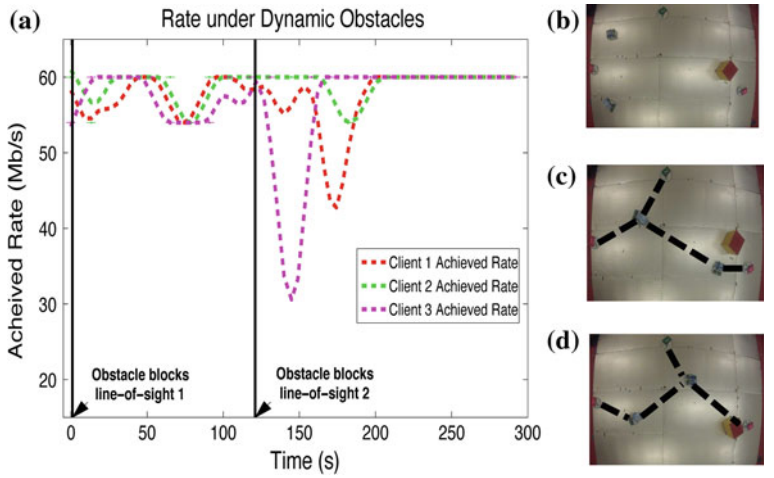


**Fig. 9** These plots show the result of disturbing the wireless channels via movement of a line-of-sight obstructing obstacle. Actual testbed snapshots are shown on the right. **b** Initial position. **c** Obstacle pos. 1. **d** Obstacle pos. 2

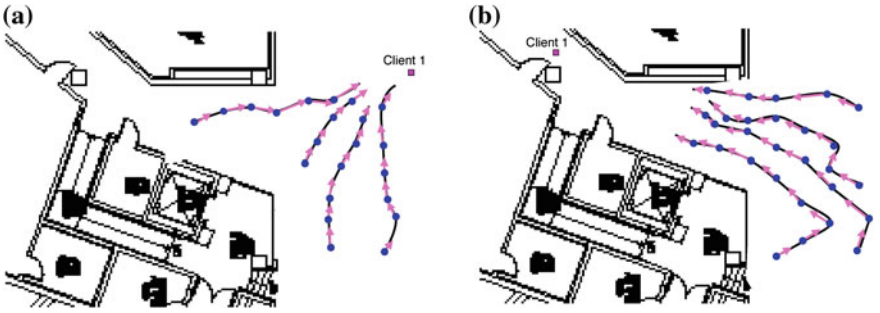


**Fig. 10** Trajectories using measured $\mathbf{v}_{\theta_{\max}}$ directions satisfy a client's demand in line-of-sight and non-line-of-sight settings in complex indoor environments. **a** Line of sight. **b** Non line of sight

**Results**: Figure 10a, b plot of candidate trajectories (from gradient field) in LOS and NLOS across initial locations. The plots show that our system successfully navigates towards the client to satisfy its demands, without knowledge of the environment or client location.

## 6 Conclusion

In this paper, we present a framework to satisfy real-time variable communication demands for a changing network. We develop a solution enabling a robotic receiver to find the profile of signal strength across spatial directions for each sender of interest.

While our technique retrieves these spatial signal profiles in real time, we note that it faces an important limitation: it assumes access to wireless channels from both the transmitter and the receiver. Developing a system that can work with unmodified transmitters remains an open challenge. Our system integrates the signal profiles with a controller that optimizes communication quality while maintaining quadratic edge costs, and thus has natural extensions to many communication-aware coordination problems such as coverage [1], consensus [3], formation control [2], etc. We believe our system provides the necessary robustness to bring the benefits of these important contributions to practical robotic systems.

**Acknowledgments** We thank Dan Feldman and Brian Julian for experimental and theoretical contributions to this work. The authors acknowledge MIT Lincoln Laboratory and MAST project under ARL Grant W911NF-08-2-0004 for their support.

## References

1. Ganguli, A., S. Susca, A., Martinez, S., Bullo, F., Cortes, J.: On collective motion in sensor networks: sample problems and distributed algorithms. In: CDC-ECC (2005)
2. Jadbabaie, A., Lin, J., Morse, A.S.: Coordination of groups of mobile autonomous agents using nearest neighbor rules. In: IEEE Transactions on Automatic Control (2003)
3. Olfati-Saber, R., Fax, J.A., Murray, R.M.: Consensus and cooperation in networked multi-agent systems. In: Proceedings of the IEEE (2007)
4. MalmirChegini, M., Mostofi, Y.: On the spatial predictability of communication channels. IEEE Trans. Wirel. Commun. **11**(3) (2012)
5. Yan, Y., Mostofi, Y.: Co-optimization of communication and motion planning of a robotic operation under resource constraints and in fading environments. IEEE Trans. Wirel. Commun. **12**(4) (2013)
6. Fink, J., Ribeiro, A., Kumar, V.: Robust control for mobility and wireless communication in cyber-physical systems with application to robot teams. In: Proceedings of the IEEE (2012)
7. Lindh, M., Johansson, K., Bicchi, A.: An experimental study of exploiting multipath fading for robot communications. In: RSS (2007)
8. Aruba Networks. Outdoor antennas and RF coverage strategies http://www.arubanetworks.com/vrd/outdoormimovrd/wwhelp/wwhimpl/common/html/wwhelp.htm#context=OutdoorMIMOVRD&file=chap4.html
9. Fitch, P.J.: Synthetic Aperture Radar. Springer, New York (1988)
10. Le Ny, J., Ribeiro, A., Pappas, G.J.: Adaptive communication-constrained deployment of mobile robotic networks. In: ACC (2012)
11. Spall, J.C.: Adaptive stochastic approximation by the simultaneous perturbation method. In: IEEE Transactions on Automatic Control (2000)
12. Kumar, S., Shi, L., Ahmed, N., Gil, S., Katabi, D., Rus, D.: Carspeak: a content-centric network for autonomous driving. SIGCOMM (2012)
13. Wang, J., Katabi, D.: Dude, where's my card?. RFID positioning that works with multipath and non-line of sight, In: SIGCOMM (2013)
14. Wang, J., Adib, F., Knepper, R., Katabi, D., Daniela, R.: Robot object manipulation using rfids. In: MobiCom, RF-Compass (2013)
15. Adib, F., Katabi, D.: See through walls with wi-fi. In: SIGCOMM (2013)
16. Halperin, D., Hu, W., Sheth, A., Wetherall, D.: Predictable 802.11 packet delivery from wireless channel measurements. In: CCR (2010)

17. Rahul, H., Kumar, S.S., Katabi, D.: Scaling wireless capacity with user demand. In: SIGCOMM, Megamimo (2012)
18. Stoica, P., Moses, R.L.: Spectral Analysis of Signals. Prentice Hall, New Jersey (2005)
19. Gil, S., Feldman, D., Rus, D.: Communication coverage for independently moving robots. In: IROS (2012)
20. Feldman, D., Gil, S., Knepper, R., Julian, B., Rus, D.: K-robots clustering of moving sensors using coresets. In: ICRA 2013 (2013)
21. Halperin, D., Hu, W., Sheth, A., Wetherall, D.: Tool release: Gathering 802.11n traces with channel state information. In: ACM SIGCOMM CCR (2011)
22. Chen, H.-C., Lin, T.-H., Kung, H.T., Lin, C.-K., Gwon, Y.: Determining RF angle of arrival using cots antenna arrays: A field evaluation. In: MILCOM (2012)
23. Xiong, J., Jamieson, K.: Arraytrack: a fine-grained indoor location system. In: NSDI (2013)
24. Joshi, K., Hong, S., Katti, S.: Pinpoint: localizing interfering radios. In: NSDI (2013)