Path Planning for a Connectivity Seeking Robot

Arjun Muralidharan and Yasamin Mostofi

***Abstract*— We consider the scenario where an unmanned vehicle needs to get connected to a remote station (or another robot). More specifically, we consider the case where an unmanned vehicle is not connected in its current location and needs to incur motion energy to find a connected spot. We are then interested in designing robot paths that are energy efficient (minimum traveled distance) and can result in guaranteed connectivity in realistic channel environments that can experience multipath, shadowing, and path loss. In this paper, we show how this problem can be optimally and efficiently solved, under mild conditions on the paths, using tools from stochastic dynamic programming. Our extensive simulations, with real channel parameters, then confirm that our approach can significantly reduce the traveled distance to connectivity, thus minimizing the total energy consumption.**

I. Introduction

Teams of unmanned vehicles have been envisioned to carry out a wide range of tasks, such as search and rescue, surveillance, exploration, sensing, and gathering information about the environment [1], [2]. In many of these scenarios, establishing connectivity with a remote station, or with another node, for the transfer of data is of paramount importance. The mobility of a robot can play an important role in enabling its connectivity as it can incur motion energy to move to spots better for connectivity. Since an unmanned vehicle has a limited energy budget, energy efficiency is another key consideration in robotic systems. In this paper, we are interested in utilizing the mobility of a robot to enable connectivity in an energy-efficient manner.

There has been considerable interest in communication-aware robotics in recent years [3]–[11]. In [6], an energy-aware trajectory is designed for a robotic network in order to enable within-network connectivity. In [7], a controller that ensures persistent intermittent connectivity is designed for a robotic network. However, oversimplified and unrealistic channel models are considered in all the above references [4]–[7]. In [8]–[11], a realistic channel model and probabilistic prediction framework based on [12] is utilized. In [8], locations of mobile robotic routers are optimized for enabling end-to-end connectivity. In [10], an optimal control-based framework is proposed to co-optimize motion and communication. In [11], the robots employ distributed beamforming, and find locations which satisfy a joint connectivity requirement, while minimizing the traveled distance. However, only final locations are optimized, and the channel quality along the path is not considered. In [13], the statistics of the distance traveled by a robot, moving along a straight line, till it gets connected is characterized.

In this paper, we are interested in the *energy-aware path planning* of a robot, to ensure reaching a *guaranteed connected spot* in a realistic channel environment that experiences multipath fading, shadowing, and path loss. To the best of our knowledge, this problem has not been optimally solved before. More precisely, we consider the problem of planning the path of a robot in order to find a connected spot while *minimizing the expected traveled distance*. We note that in a realistic channel environment, the robot's knowledge of the connectivity at any location is stochastic [12]. Hence, the traveled distance till connectivity is not known a priori, and is a random variable. Our objective is then to find a path that minimizes the expected traveled distance till connectivity. Fig. 1 shows an example of this scenario. In this paper, we show how this problem can be optimally and efficiently solved, under mild conditions on the paths, using tools from stochastic dynamic programming.

The paper is organized as follows. In Section II, we formally introduce our problem and review relevant literature on channel modeling and probabilistic channel prediction. In Section III, we show how to optimally design the path, using a stochastic dynamic programming framework, where we place a mild constraint on the solution. Finally, we confirm the performance and efficiency of our approach with extensive simulations using real channel parameters, in Section IV.

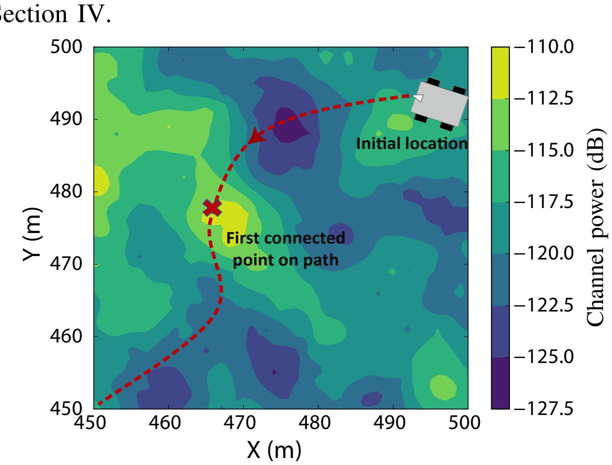

Fig. 1: The robot moves along a path until it gets connected. The background color at each location represents the corresponding channel power when transmitting from that location. The objective is to design a path that minimizes the expected traveled distance till connectivity.

II. Problem Setup

In this section, we formally define our problem of interest. We first provide an overview of probabilistic channel model-

This work is supported in part by NSF NeTS award 1321171 and NSF RI award 1619376.

The authors are with the Department of Electrical and Computer Engineering, University of California Santa Barbara, Santa Barbara, CA 93106, USA email: {arjunm, ymostofi}@ece.ucsb.edu.

ing and prediction [12], [14]. We then introduce our problem of interest: **Planning the optimum path for the robot that minimizes the expected distance till connectivity.** Our main contribution is proposing a computationally efficient approach that can find the optimal path, under mild conditions, using tools from stochastic dynamic programming literature.

A. Channel Modeling [12], [14]

A communication channel is best modeled as a multi-scale random process with three major components: path loss, shadowing and multipath fading [12], [14]. Let $\Gamma_{\text{dB}}(q_1)$ denote the received channel power (in dB) from a transmitter at location $q_1 \in \mathbb{R}^2$ to the remote station located at the origin. It can then be expressed as $\Gamma_{\text{dB}}(q_1) = \Gamma_{\text{PL,dB}}(q_1) + \Gamma_{\text{SH,dB}}(q_1) + \Gamma_{\text{MP,dB}}(q_1)$ where $\Gamma_{\text{SH,dB}}(q_1)$ and $\Gamma_{\text{MP,dB}}(q_1)$ are random variables denoting the impact of shadowing and multipath respectively, and $\Gamma_{\text{PL,dB}}(q_1) = K_{\text{dB}} - 10 n_{\text{PL}} \log_{10} \|q_1\|$ is the distance-dependent path loss with n_{PL} representing the path loss exponent. $\Gamma_{\text{SH,dB}}(q_1)$ is best modeled as a Gaussian random variable with an exponential spatial correlation: $\mathbb{E}\left\{\Gamma_{\text{SH,dB}}(q_1)\Gamma_{\text{SH,dB}}(q_2)\right\} = \sigma_{\text{SH}}^2 e^{-\|q_1 - q_2\|/\beta_{\text{SH}}}$ where σ_{SH}^2 is the shadowing power and β_{SH} is the decorrelation distance.

The shadowing or large-scale fading component is the result of attenuation through large stationary objects in the environment, such as buildings or hills. We thus assume that the path loss and shadowing component stay constant with time. Multipath fading or small scale fading, on the other hand, is the result of the additions of multiple paths. Thus, small changes in the positions of reflectors and scatterers in the environment can cause large changes in the multipath fading signal. To account for this, we then assume that the multipath component is time-varying. This then implies that the multipath value at any location during the operation phase of the robot, is independent of its corresponding value when the prior measurements were collected [15]–[17].

We next briefly describe the probabilistic channel prediction framework of [12], which is based on a few prior channel measurements in the workspace. These measurements could have been collected by the same robot during prior operations, or collected by other robots in the past.

B. Realistic Channel Prediction [12], [14]

Let $\Gamma_{q,\text{dB}}$ represent the vector of m a priori-gathered received channel power measurements (in dB) in the same environment, where $q = [q_1 \cdots q_m]^{\text{T}}$ denotes the vector of the corresponding positions.

Lemma 1 (See [12] for proof): A Gaussian random vector, $\Gamma_{\text{dB}}(r) = [\Gamma_{\text{dB}}(r_1) \cdots \Gamma_{\text{dB}}(r_k)]^{\text{T}} \sim \mathcal{N}\left(\overline{\Gamma}_{\text{dB}}(r), \Sigma_{\text{dB}}(r)\right)$ can best characterize the vector of channel power (in the dB domain) when transmitting from locations $r = [r_1 \cdots r_k]^{\text{T}}$, with the mean and covariance given by

$$\overline{\Gamma}_{\text{dB}}(r) = H_r \hat{\vartheta} + \Psi_{r,q} \Phi_q^{-1}\big(\Gamma_{q,\text{dB}} - H_q \hat{\vartheta}\big),$$
$$\Sigma_{\text{dB}}(r) = \Phi_r - \Psi_{r,q} \Phi_q^{-1} \Psi_{r,q}^{\text{T}},$$

where $\hat{\vartheta} = [\hat{K}_{\text{dB}}, \hat{n}_{\text{PL}}]^T$, $H_r = [\mathbf{1}_k \quad - D_r]$, $H_q = [\mathbf{1}_m \quad - D_q]$, $\mathbf{1}_m$ ($\mathbf{1}_k$) represents the m-dimensional (k - dimensional) vector of all ones, $D_q = \left[10\log_{10}(\|q_1\|) \ \cdots \ 10\log_{10}(\|q_m\|)\right]^{\text{T}}$ and $D_r = \left[10\log_{10}(\|r_1\|) \ \cdots \ 10\log_{10}(\|r_k\|)\right]^{\text{T}}$. Furthermore, Φ_q, Φ_r and $\Psi_{r,q}$ denote matrices with entries $\left[\Phi_q\right]_{i_1,i_2} = \hat{\sigma}_{\text{SH}}^2 e^{-\|q_{i_1} - q_{i_2}\|/\hat{\beta}_{\text{SH}}} + \hat{\sigma}_{\text{MP}}^2 \, \delta_{i_1,i_2}$, $\left[\Phi_r\right]_{j_1,j_2} = \hat{\sigma}_{\text{SH}}^2 e^{-\|r_{j_1} - r_{j_2}\|/\hat{\beta}_{\text{SH}}} + \hat{\sigma}_{\text{MP}}^2 \delta_{j_1,j_2}$ and $\left[\Psi_{r,q}\right]_{j_1,i_1} = \hat{\sigma}_{\text{SH}}^2 \ e^{-\|r_{j_1} - q_{i_1}\|/\hat{\beta}_{\text{SH}}}$, where $i_1, i_2 \in \{1, \cdots, m\}$, $j_1, j_2 \in \{1, \cdots, k\}$ and $\delta_{i,j} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{else} \end{cases}$.
The $\hat{}$ symbol denotes the estimate of the corresponding parameter. These underlying channel parameters are estimated based on the a priori measurements as well. See [12] for more details on this channel prediction framework, as well as its performance with real data and in different environments. It should be noted that [12] focuses on the case where the multipath component is time-invariant. It can be easily extended to account for the case where the multipath component of the operation phase is independent of its corresponding prior measurement value, as mentioned earlier in Section II-A.

We next describe our problem of interest and show how it can be posed as a graph-theoretic problem. The aforementioned channel prediction framework will then be used to estimate the probability of connectivity at any location in the workspace.

C. Minimizing the Expected Distance till Connectivity

Consider the scenario of a robot that does not have connectivity to a remote station at its current location and thus needs to find a connected location. In order to do so, the transmission would have to satisfy a Quality of Service (QoS) requirement, such as a target bit error rate (BER), which would in turn imply a minimum required received channel power given a fixed transmit power. Thus, in order for the robot to successfully establish connectivity, it needs to find a location where the channel power in transmission from that location would be greater than the minimum required channel power. As shown in Section II-B, the robot's knowledge of the channel is stochastic. Its goal is thus to plan a path such that it minimizes the expected traveled distance till connectivity.

We next describe how to pose this in a graph-theoretic framework. We first represent the workspace as an undirected graph. More specifically, we discretize the workspace of the robot into cells, where each cell serves as a node in the graph. Each cell is assigned a probability of connectivity that is estimated by the robot based on the channel prediction framework of Section II-B. A cell is said to be connected if there exists a location in the cell that is connected. For instance, consider a cell that consists of locations $r = [r_1, \cdots, r_k]^T$. The probability of failure of connectivity of the cell is then given by $\Pr(\Gamma_{\text{dB}}(r_i) < \Gamma_{\text{th,dB}}, \forall i \leq k)$, where $\Gamma_{\text{dB}}(r) = [\Gamma_{\text{dB}}(r_1) \cdots \Gamma_{\text{dB}}(r_k)]^{\text{T}} \sim \mathcal{N}\left(\overline{\Gamma}_{\text{dB}}(r), \Sigma_{\text{dB}}(r)\right)$ is a Gaussian random vector, as described in Section II-B, and $\Gamma_{\text{th,dB}}$ is the minimum required channel power for connectivity.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be the undirected graph constructed from the workspace, where $\mathcal{V}$ denotes the set of nodes and $\mathcal{E}$ denotes the set of edges. Let $p_v \in [0,1]$ be the probability of connectivity at node $v \in \mathcal{V}$, and let d_{uv} denote the weight of the edge $(u,v) \in \mathcal{E}$. The edge weight d_{uv} is the minimum physical distance between the nodes u and v. The connectivity of a node is independent of the connectivity of the other nodes in the graph.[1] Let $v_s \in \mathcal{V}$ denote the starting node of the robot. The objective is to produce a path starting from node v_s that *minimizes the expected traveled distance till connectivity*. The optimization is then expressed as:

$$\begin{aligned} \underset{\mathcal{P}}{\text{minimize}} \quad & \mathbb{E}\left[\text{traveled dist on } \mathcal{P}\right] \\ \text{subject to} \quad & \text{Connected} \\ & \mathcal{P} \text{ is a path on } \mathcal{G} \\ & \mathcal{P}[1] = v_s. \end{aligned} \tag{1}$$

In other words, the average traveled distance on the optimal path till the robot gets connected is smaller than the average distance till connectivity on any other possible path on the graph. Note that the robot may only traverse part of the entire path produced by its planning, as its planning is based on stochastic knowledge, and connectivity may occur at any point along the path.

For the expected distance till connectivity of a path to be well defined, the probability of not being connected after traversing the path must be 0. This implies that the path's final node must be one where connectivity is guaranteed, i.e., a v such that $p_v = 1$. We call such a node a *terminal* node and let $T = \{v \in \mathcal{V} : p_v = 1\}$ denote the set of terminal nodes. We assume that the set T is non-empty, i.e., there exists at least one node in the workspace that is guaranteed to be connected. This is a valid assumption since the robot will get connected if it gets close enough to the remote station. A feasible path thus has a terminal node as its end node. Fig. 2 shows a toy example as well as a feasible solution path.

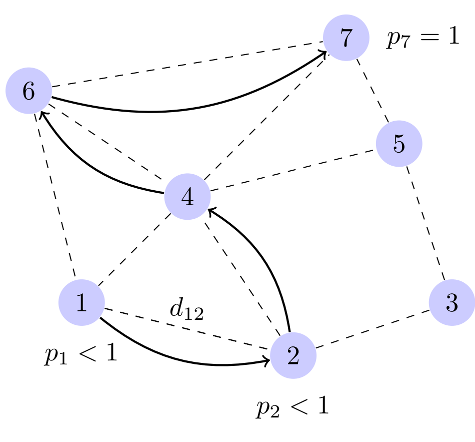

Fig. 2: A toy example along with a feasible solution path starting from node 1.

We next characterize the expected distance first for paths where nodes are not revisited, and then generalize it to all possible paths. Let the path $\mathcal{P} = (v_1, v_2, \cdots, v_m = t)$ be a sequence of m nodes such that no node is revisited, i.e., $v_i \neq v_j,\ \forall j \neq i$, and which terminates at a terminal node $t \in T$. Let $C(\mathcal{P}, i)$ represent the expected traveled distance till connectivity, from node $\mathcal{P}[i] = v_i$ onward. We can write $C(\mathcal{P}, 1)$ as follows,

$$\begin{aligned} C(\mathcal{P},1) =& p_{v_1} \times 0 + (1-p_{v_1}) p_{v_2} d_{v_1 v_2} + \cdots \\ & + \left[\prod_{j \le m-1} (1-p_{v_j})\right] p_{v_m} (d_{v_1 v_2} + \cdots + d_{v_{m-1} v_m}) \\ =& (1-p_{v_1}) d_{v_1 v_2} + (1-p_{v_1})(1-p_{v_2}) d_{v_2 v_3} + \cdots \\ & + \left[(1-p_{v_1}) \cdots (1-p_{v_{m-1}})\right] d_{v_{m-1} v_m} \\ =& \sum_{i=1}^{m-1} \left[\prod_{j \le i} (1-p_{v_j})\right] d_{v_i v_{i+1}}. \end{aligned} \tag{2}$$

For a path which contains revisited nodes, the expected distance can then be given by

$$\begin{aligned} C(\mathcal{P},1) &= \sum_{i=1}^{m-1} \left[\prod_{j \le i : v_j \neq v_k, \forall k < j} (1-p_{v_j})\right] d_{v_i v_{i+1}} \\ &= \sum_{e \in \mathcal{E}(\mathcal{P})} \left[\prod_{v \in \mathcal{V}(\mathcal{P}_e)} (1-p_v)\right] d_e, \end{aligned}$$

where $\mathcal{E}(\mathcal{P})$ denotes the set of edges belonging to the path $\mathcal{P}$, and $\mathcal{V}(\mathcal{P}_e)$ denotes the set of vertices encountered along $\mathcal{P}$ until the edge $e \in \mathcal{E}(\mathcal{P})$. Note that this expression becomes the same as (2), when there are no revisited nodes in the path.

Moreover, $C(\mathcal{P}, i)$ can be expressed recursively as

$$C(\mathcal{P},i) = \begin{cases} (1-p_{v_i})\left(d_{v_i v_{i+1}} + C(\mathcal{P}, i+1)\right), & \\ & \text{if } v_i \neq v_k, \forall k < i \\ d_{v_i v_{i+1}} + C(\mathcal{P}, i+1), & \text{else} \end{cases}. \tag{3}$$

We utilize this recursive expression in the dynamic programming formulation of the next section.

The optimization of (1) can then be formally expressed as

$$\begin{aligned} \underset{\mathcal{P}}{\text{minimize}} \quad & C(\mathcal{P}, 1) \\ \text{subject to} \quad & \mathcal{P} \text{ is a path on } \mathcal{G} \\ & \mathcal{P}[1] = v_s \\ & \mathcal{P}[\text{end}] \in T. \end{aligned} \tag{4}$$

III. Stochastic Dynamic Programming Solution

Finding the solution to the problem formulation of (4) efficiently is challenging in general. One can show that (4) can be posed in a stochastic dynamic programming framework. However, the resulting state space is exponential in the number of nodes of the graph, and it is thus not feasible to solve efficiently. In this section, we place a mild requirement that a feasible solution path must satisfy. We then show how to find the optimal path efficiently among the

[1]This assumption is only for the purpose of mathematical derivations. When testing our proposed approach in Section IV, we consider realistic channel realizations which may result in a spatially correlated connectivity, depending on the environment.

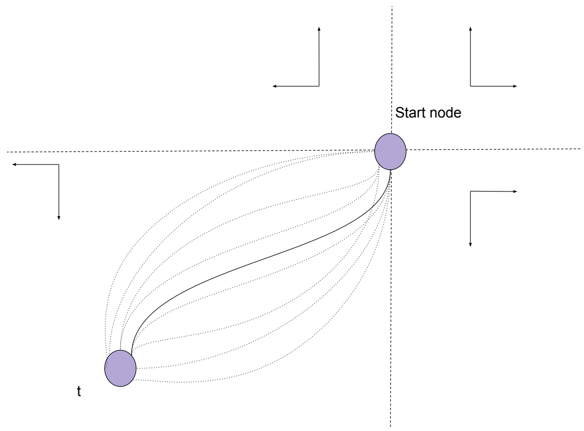

Fig. 3: A DAG is imposed which only allows "outward" motion from the starting node.

set of paths that satisfy this requirement using a stochastic dynamic programming formulation.

More specifically, we consider the following requirement that a feasible solution must satisfy: *Each successive node on the path must be further away from the starting node v_s.* This implies that for a path $\mathcal{P} = (v_1 = v_s, v_2, \cdots, v_m)$, the condition $d^{\min}_{v_s v_i} > d^{\min}_{v_s v_{i-1}}$ must be satisfied, for all i, where $d^{\min}_{v_s v_i}$ is the shortest distance between v_s and v_i on the graph. In the case of a grid graph with a single terminal node, this implies that a path must always move towards the terminal node, which is a reasonable requirement to impose.

It can be seen that imposing this requirement is equivalent to imposing a directed acyclic graph (DAG) on the original graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. The imposed DAG, $\mathcal{G}_{\text{DAG}} = (\mathcal{V}, \mathcal{E}_{\text{DAG}})$, has the same set of nodes $\mathcal{V}$, and the set of edges is given by $\mathcal{E}_{\text{DAG}} = \{(u,v) \in \mathcal{E} : d^{\min}_{v_s v} > d^{\min}_{v_s u}\}$ where (u,v) represents a directed edge from u to v. As a concrete example consider an $n \times n$ grid workspace, where neighboring cells are limited to {left, right, top, down} cells. This would result in an $n \times n$ DAG. The resulting DAG would be such that only outward flowing edges from the start node are allowed. In other words, the start node v_s serves as the center. We then form outward moving edges for each quadrant. In the first quadrant only right and upward edges are allowed, in the second quadrant only left and upward edges are allowed, and so on. Fig. 3 shows an example of this, where several feasible paths from the start node to a terminal node are shown.

A. Proposed Path Planning Strategy

We next show that the problem of finding the path on $\mathcal{G}_{\text{DAG}}$, with a minimum expected distance, can be formulated as an infinite horizon Markov Decision Process (MDP) with an absorbing state, a formulation known in the stochastic dynamic programming literature as the *stochastic shortest path* (SSP) problem [18]. In this part, we first provide a brief summary of the SSP formulation. We then show how our problem of finding a shortest path to connectivity can be formulated as an SSP, which can then be efficiently solved, as we shall see.

The stochastic shortest path problem (SSP) is a general mathematical formulation in dynamic programming [18], which is specified by a state space S, control/action constraint sets $A(s)$ for $s \in S$, state transition probabilities $P_{ss'}(a) = \Pr(s_{k+1} = s' | s_k = s, a_k = a)$, an absorbing termination state $s_t \in \mathcal{S}$, and a cost function $g(s,a)$ for $s \in S$ and $a \in A(s)$. The goal is then to obtain a policy that would lead to the terminal state s_t with the probability of 1, and with minimum expected cost.

We next show how our path optimization problem of (4) on $\mathcal{G}_{\text{DAG}}$ can fit into an SSP formulation. Since we've imposed a DAG, any path on $\mathcal{G}_{\text{DAG}}$ cannot contain revisited nodes. Then, the recursive expression of (3) for the expected traveled distance on $\mathcal{P}$ from node $\mathcal{P}[i] = v_i$ becomes $C(\mathcal{P}, i) = (1 - p_{v_i})\big(d_{v_i v_{i+1}} + C(\mathcal{P}, i+1)\big)$, i.e., it can be expressed in terms of the expected distance of the neighboring node that the path visits next. Thus, we can see that the optimal path from a node v, would also contain the optimal path from the neighboring node that the path visits next. This motivates the use of a stochastic dynamic programming framework, where the expected cost to go $J(v)$, of a state v, represents the expected traveled distance to connectivity from v.

More precisely, we formulate the SSP as follows. Let $\mathcal{V}' = \mathcal{V} \setminus T$ be the set of non-terminal nodes in the graph. The current node $v \in \mathcal{V}'$ of the robot is a state of the SSP. The state space is thus given by $S = \mathcal{V}' \cup \{s_t\}$ where s_t denotes the absorbing termination state. In this setting, s_t denotes the state that the robot is connected. The termination state s_t is absorbing, i.e., $P_{s_t s_t}(a) = 1, \forall a \in A(s_t)$. The action set $A(v)$ available at state v is given by the set of its neighbors, i.e., $A(v) = \{u \in \mathcal{V} : (v,u) \in \mathcal{E}_{\text{DAG}}\}$. The state transition probability given that neighbor a is the selected action is given as,

$$P_{vu}(a) = \begin{cases} 1 - p_v, & \text{if } u = f(a) \\ p_v, & \text{if } u = s_t \\ 0, & \text{else} \end{cases},$$

where $f(a) = \begin{cases} a, & \text{if } a \in \mathcal{V}' \\ s_t, & \text{if } a \in T \end{cases}$. This implies that at node v, with probability p_v, the robot will get connected. If not, it will move to the selected neighbor a. The cost incurred when neighbor $a \in A(s)$ is selected at node v is given by $g(v,a) = (1 - p_v) d_{va}$, i.e., the expected traveled distance from v to a accounting for the fact that the robot can get connected at v. This stochastic shortest path problem formulation is different from a traditional shortest path problem.

The minimum expected distance when starting from state v_0 is then given by

$$J^*(v_0) = \min_{\mu} \mathop{\mathbb{E}}_{\{v_k\}} \left[\sum_{k=0}^{\infty} g(v_k, \mu(v_k)) \right],$$

where μ is a policy that prescribes what action to take at a given state, i.e., $\mu(v)$ is the action to take at state v. The policy μ specifies which neighboring node to move to next, i.e., if at node v, then $\mu(v)$ is the next node to go to. The objective here is to find the optimal policy μ^*, that would minimize the expected distance on the imposed DAG. Note that given μ^*, we can obtain the optimal solution path $\mathcal{P}^* = (v_1 = v_s, v_2, \cdots, v_m)$, where $v_{k+1} = \mu^*(v_k)$, for all k, and $v_m \in T$.

We next show, in the following Lemma, that the optimal solution can be characterized by the Bellman equation.

Lemma 2: The optimal cost function J^* is the unique solution of the Bellman equation:

$$J^*(v) = \min_{u \in A(v)} \{(1 - p_v) d_{vu} + (1 - p_v) J^*(f(u))\},$$

and the optimal policy μ^* is given by

$$\mu^*(v) = \arg \min_{u \in A(v)} \{(1 - p_v) d_{vu} + (1 - p_v) J^*(f(u))\},$$

for all $v \in \mathcal{V}'$ and where $J^*(s_t) = 0$.

Proof: It can be confirmed that the formulation satisfies Assumption 2.1.1 and 2.1.2 in [18]. The proof is then provided in [18]. ■

The optimal solution with the minimum expected distance can then be found by running value iteration [18]:

$$J_{k+1}(v) = \min_{u \in A(v)} \{(1 - p_v) d_{vu} + (1 - p_v) J_k(f(u))\},$$

with the policy at iteration $k+1$ given by

$$\mu_{k+1}(v) = \arg \min_{u \in A_v} \{(1 - p_v) d_{vu} + (1 - p_v) J_k(f(u))\},$$

for all $v \in \mathcal{V}'$ and where $J_k(s_t) = 0$, for all k.

We next discuss the computational complexity of utilizing value iteration and show how it can efficiently provide us with the optimal solution. In the following Lemma, we show that it converges to the optimal solution in at most the same number of iterations as there are nodes in the graph.

Lemma 3 (Computational complexity): When we start from $J_0(v) = \infty$, for all $v \in \mathcal{V}'$, the value iteration method yields the optimal solution after at most $|\mathcal{V}'|$ iterations.

Proof: This follows from the convergence analysis of value iteration on a SSP problem with a DAG structure, provided in [18]. ■

IV. Numerical Results

Consider a scenario where a robot is located in a 50 m × 50 m workspace with the remote station at the origin as shown in Fig. 4. The initial location of the robot is at the corner of the workspace as shown in Fig. 4. The channel is generated using the probabilistic channel model described in Section II-A, with the following parameters that were obtained from real channel measurements in downtown San Francisco [15] : $n_{\text{PL}} = 4.2$, $\sigma_{\text{SH}} = 2.9$ and $\beta_{\text{SH}} = 12.92$ m. Moreover, the multipath fading is taken to be uncorrelated Rician fading with the parameter $K_{\text{ric}} = 1.59$. In order for the robot to be connected, a minimum received power of $P_{R,\text{th,dBm}} = -80$ dBmW is required, and the maximum transmission power of the robot is taken to be $P_{0,\text{dBm}} = 27$ dBmW [19].

The robot is assumed to have 5 % a priori measurements in the workspace. It then utilizes the channel prediction framework described in Section II-B to predict the channel

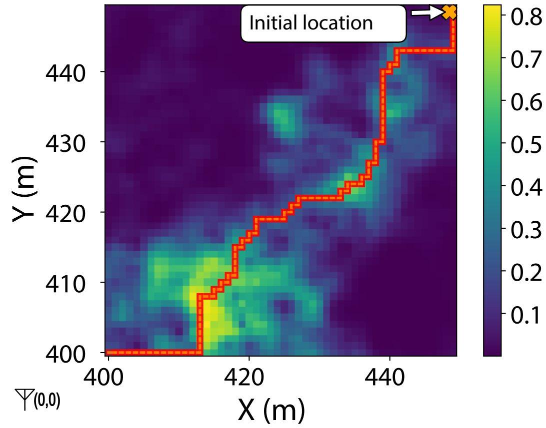

Fig. 4: The optimum path based on our proposed approach for a channel realization. The background color plot denotes the predicted probability of connectivity, which is used by the robot for path planning. Readers are referred to the color pdf for better visibility.

at any unvisited location. We discretize the workspace into cells of dimension 1 m × 1 m, and each such cell serves as a node in the graph. The actions available to the robot at every node are {left, right, top, down}. This gives us a grid graph of dimension 50×50. We obtain an estimated probability of connectivity for every node, as described in Section II-C, which will be utilized by the robot for its path planning. We also add a new terminal node in the graph with the probability of connectivity 1, which represents the remote station at the origin. The node closest to the remote station in the workspace is attached to this new terminal node with a edge weight equal to the expected distance till connectivity when moving straight towards the remote station from the node. This can be calculated based on the work in [13].

	Our approach	Baseline heuristic
Avg distance (m)	37.33±24.10	67.63±40.89

TABLE I: The average traveled distance along with the corresponding standard deviation, for our proposed approach and for the baseline heuristic approach. The average is obtained by averaging over 500 channel realizations. We can see that our approach results in a significant reduction in the traveled distance.

We next compare our proposed approach with a baseline heuristic of moving straight towards the remote station. The performance of the approaches are evaluated based on the true probability of connectivity of a node calculated based on the true value of the channel. Fig. 4 shows the solution path produced by our proposed approach for a sample channel realization. The background plot denotes the robot's prediction of the probability of connectivity in the workspace. We see that the optimum path takes a detour, from its route to the remote station, in order to visit areas of good probability of connectivity. Table I shows the expected distance along with the corresponding standard deviation, for our proposed approach and the baseline heuristic, averaged over 500 channel realizations. We see that the our approach

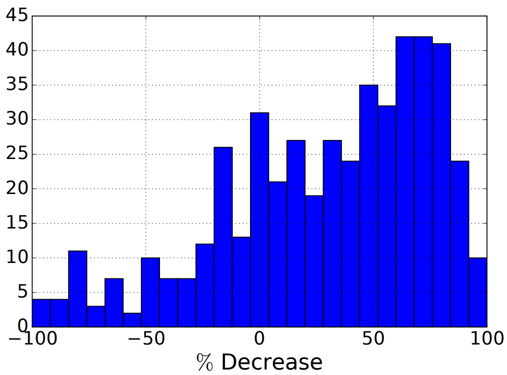

Fig. 5: Histogram of the percentage decrease in the traveled distance, when using our approach as compared to the baseline heuristic. The data is obtained by using 500 channel realizations.

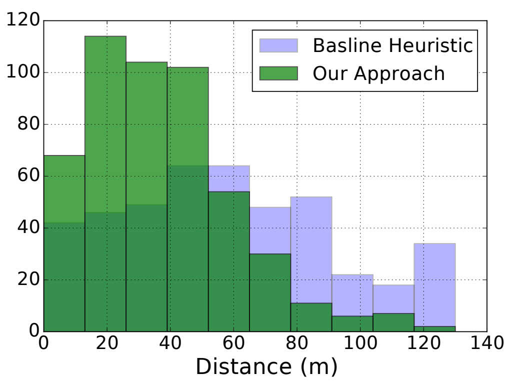

Fig. 6: Histogram of the average distance for our proposed approach and the baseline heuristic, over 500 channel realizations.

outperforms the baseline heuristic significantly and provides an overall 45% reduction in the expected traveled distance. Fig. 5 shows the distribution of the percentage decrease in the traveled distance, obtained by using our approach instead of the baseline heuristic. As can be seen, our proposed approach can result in a significant reduction in the traveled distance. In a small fraction of the cases however, there is an increase in the expected cost when using our approach. This arises due to a mismatch between the true and the predicted probability of the connectivity map. Finally, Fig. 6 shows the histogram of the traveled distance using 500 channel realizations. We can see that our approach can reduce the traveled distance and thus the total motion energy consumption significantly.

V. Conclusions

In this paper, we considered the scenario where an unmanned vehicle, that is not currently connected, needs to utilize its mobility to find a connected spot to a remote station (or another unmanned vehicle), in an energy efficient way. More specifically, we considered a robotic path planning problem in realistic communication environments (e.g., multipath, shadowing, path loss), where the robot needs to find a connected spot while minimizing its expected traveled distance. We showed how this problem can be posed in a graph-theoretic framework. Then, by utilizing tools from the stochastic dynamic programming literature, we showed how it is possible to obtain the optimal solution to this challenging problem, by putting a mild condition on the paths. Finally, we confirmed the efficiency of our proposed approach with extensive simulations, using channel parameters obtained from real channel measurements. Our results showed a significant saving in the traveled distance when compared to baseline heuristic approaches.

References

[1] Alireza Ghaffarkhah and Yasamin Mostofi. Path planning for networked robotic surveillance. *IEEE Transactions on Signal Processing*, 60(7):3560–3575, 2012.

[2] Ali Marjovi, João Gonçalo Nunes, Lino Marques, and Aníbal de Almeida. Multi-robot exploration and fire searching. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1929–1934, 2009.

[3] Magnus Lindhé and Karl Henrik Johansson. Using robot mobility to exploit multipath fading. *IEEE Wireless Communications*, 16(1):30–37, 2009.

[4] Cory Dixon and Eric W Frew. Optimizing cascaded chains of unmanned aircraft acting as communication relays. *IEEE Journal on Selected Areas in Communications*, 30(5):883–898, 2012.

[5] Chia Ching Ooi and Christian Schindelhauer. Minimal energy path planning for wireless robots. *Mobile Networks and Applications*, 14(3):309–321, 2009.

[6] Ran Dai, Joshua Maximoff, and Mehran Mesbahi. Optimal trajectory generation for establishing connectivity in proximity networks. *IEEE Transactions on Aerospace and Electronic Systems*, 49(3):1968–1981, 2013.

[7] Yiannis Kantaros and Michael M Zavlanos. Distributed intermittent connectivity control of mobile robot networks. *IEEE Transactions on Automatic Control*, 62(7):3109–3121, 2017.

[8] Yuan Yan and Yasamin Mostofi. Robotic router formation in realistic communication environments. *IEEE Transactions on Robotics*, 28(4):810–827, 2012.

[9] Yuan Yan and Yasamin Mostofi. Co-optimization of communication and motion planning of a robotic operation under resource constraints and in fading environments. *IEEE Transactions on Wireless Communications*, 12(4):1562–1572, 2013.

[10] Usman Ali, Hong Cai, Yasamin Mostofi, and Yorai Wardi. Motion and communication co-optimization with path planning and online channel prediction. In *American Control Conference (ACC)*, pages 7079–7084, 2016.

[11] Arjun Muralidharan and Yasamin Mostofi. Distributed beamforming using mobile robots. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6385–6389, 2016.

[12] Mehrzad Malmirchegini and Yasamin Mostofi. On the spatial predictability of communication channels. *IEEE Transactions on Wireless Communications*, 11(3):964–978, 2012.

[13] Arjun Muralidharan and Yasamin Mostofi. First passage distance to connectivity for mobile robots. In *American Control Conference (ACC)*, pages 1517–1523, 2017.

[14] Alireza Ghaffarkhah and Yasamin Mostofi. Channel learning and communication-aware motion planning in mobile networks. In *American Control Conference (ACC)*, pages 5413–5420, 2010.

[15] William M Smith and Donald C Cox. Urban propagation modeling for wireless systems. Technical report, DTIC Document, 2004.

[16] Ravi Narasimhan and Donald C Cox. A handoff algorithm for wireless systems using pattern recognition. In *the Ninth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications,*, volume 1, pages 335–339, 1998.

[17] Ravi Narasimhan and Donald C Cox. Estimation of mobile speed and average received power with application to corner detection and handoff. In *the 33rd Asilomar Conf. on Signals, Systems & Computers*. Citeseer, 1999.

[18] Dimitri P Bertsekas. *Dynamic programming and optimal control*, volume 1. Athena Scientific Belmont, MA, 1995.

[19] S Lönn, U Forssen, P Vecchia, A Ahlbom, and M Feychting. Output power levels from mobile phones in different geographical areas; implications for exposure assessment. *Occupational and Environmental Medicine*, 61(9):769–772, 2004.