ICC 2021 - IEEE International Conference on Communications | 978-1-7281-7122-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICC42927.2021.9500912

Communication-Aware RRT*: Path Planning for Robotic Communication Operation in Obstacle Environments

Winston Hurst, Hong Cai, and Yasamin Mostofi

***Abstract*— In this paper, we are interested in path optimization for robotic communication operations in obstacle environments. Consider a robot that needs to perform a given communication task (e.g., data uploading, broadcasting, or relaying) while navigating from a start position to a designated final position, avoiding obstacles, and minimizing its total motion and communication costs. Our goal is to develop a general path planning algorithm applicable to various robotic communication scenarios in which a robot operates in realistic channel fading environments and in the presence of obstacles. We show how we can adapt the traditional Rapidly-Exploring Random Tree Search Star (RRT*) path planning algorithm to jointly consider both communication and motion objectives in realistic channel environments that contain obstacles. We further show that our proposed approach can provide theoretical optimality guarantees while being computationally efficient. We extensively evaluate our proposed approach in realistic wireless channel environments for various transmission settings and communication tasks. The results demonstrate the efficacy of our proposed approach.**

I. INTRODUCTION

Recent years have seen great progress in mobile robotics, creating new opportunities for wireless communications. Unmanned vehicles, for instance, can use their mobility to enable connectivity in areas that would otherwise be poorly connected [1]. On the other hand, robust communication is also crucial for any networked robotic operation to ensure proper information flow and task completion. Enabling this vision of robust networked robotics requires a joint consideration of both communication and motion aspects of a robotic operation, a field referred to as *communication-aware robotics* [2], [3].

Earlier work in this area from the robotics community used over-simplified communication models, e.g., the disk/path-loss model, which can lead to poor performance in real-world wireless channel environments. In recent years, researchers have started to incorporate realistic wireless fading models when optimizing the robot's motion [4]–[9], properly co-optimizing the motion and communication parts of the operation. See [3] for a recent review of the state-of-the-art in mobility-enabled connectivity.

However, due to the complex spatial dynamics of wireless channels, the need for predicting channel quality at unvisited locations, and the high dimensionality of path planning, it is challenging to optimally plan motion and communication jointly. To make the problem more tractable, existing work frequently resorts to heuristics which may not have optimality guarantees [4]–[7]. In particular, while obstacle avoidance is at the heart of any robotic operation and is extensively addressed in the robotics literature, it is not considered in most communication-aware robotics papers that attempt to address operations in realistic fading environments.

Winston Hurst, Hong Cai, and Yasamin Mostofi are with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, USA (email: {winstonhurst, hcai, ymostofi}@ece.ucsb.edu). This work was supported in part by NSF CCSS award 1611254 and NSF RI award 2008449.

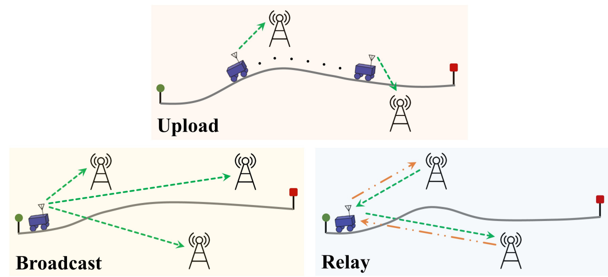

Fig. 1: The robot performs a given communication task, such as uploading, broadcasting, and relaying, while navigating from the start position (green disk) to the given destination (red square).

This is the main motivation for this paper, to consider networked robotic operations in real channel environments that contain obstacles. More specifically, we show how we can adapt the Rapidly-Exploring Random Tree Search Star (RRT*) path planning framework [10] to co-optimize motion and communication objectives in realistic channel environments that contain obstacles. RRT* is an efficient path planning framework that can handle arbitrarily-located obstacles and provide asymptotic performance guarantees, but it is traditionally used for motion objectives without considering communication.

In this paper, we are interested in a variety of robotic communication operations. Specifically, consider the case where the robot navigates from a start position to a destination while given a communication task, as shown in Fig. 1. First, we consider a data uploading operation where the robot needs to upload its onboard data (e.g., in a surveillance mission) to a remote station as it traverses the field. In the second case, the robot needs to broadcast data to a number of remote nodes in the field. Lastly, in the third scenario, the robot is tasked with relaying data between two remote nodes that are otherwise disconnected, while traveling from its start position to the final position.

While these different scenarios entail different cost functions, we show how to pose a unifying path-communication optimization framework. We then show how we can adapt RRT* to solve the co-optimization problem, while providing obstacle avoidance and considering realistic fading environments. As we shall see, our cost functions satisfy an additivity requirement, which allows our adapted RRT* approach to generate paths with asymptotic optimality.

978-1-7281-7122-7/21/$31.00 ©2021 IEEE

II. System Modeling

In this section, we summarize the communication and motion models used in this paper.

A. Communication and Channel Prediction Model

Consider the case where the robot needs to transmit data to a remote station. The received Signal-to-Noise Ratio (SNR) at the station is given by $SNR_{\text{rec}} = \Gamma_T \cdot \Upsilon$, where Γ_T is the robot's transmission power and Υ is the received Channel-to-Noise Ratio (CNR). The CNR varies spatially due to path loss, shadowing, and multipath fading. A Quality of Service (QoS) requirement, such as a target Bit Error Rate (BER), will result in a minimum required received SNR of a transmission, SNR_{th}. If $SNR_{\text{rec}} \geq SNR_{\text{th}}$, then the robot can communicate with the remote station while satisfying the QoS requirement, and we say the robot is *connected*. Otherwise, we say it is *disconnected*.

If the robot cannot change its transmission power, the SNR requirement translates to a minimum required CNR, Υ_{th}. In this case, the spatially-varying CNR directly dictates the robot's connectivity. Alternatively, consider the case where the robot can adapt its transmission power. Suppose the robot adopts the commonly-used MQAM modulation for transmission. The required transmission power is well approximated by $\Gamma_T = (2^r - 1)\ln(5p_{\text{BER}})/(-1.5\Upsilon)$, where r is the spectral efficiency and p_{BER} is the required BER. Given the QoS requirements (e.g., BER and spectral efficiency), it can be seen that the robot's transmission power also depends on the spatially-varying CNR.

For the purpose of path planning, the robot needs to assess its connectivity (or required transmission power) at unvisited locations, which requires a prediction of the CNR over the space. Reference [11] shows how the robot can make such a prediction based on a spatial stochastic process model of the CNR, which accounts for the real-world propagation effects of path loss, shadowing, and multipath fading. More specifically, given a small number of prior channel samples in the same environment, the CNR (in dB) at an unvisited location x, $\Upsilon_{\text{dB}}(x)$, can be best modeled by a Gaussian random variable, with its expectation and variance given as follows:

$$\begin{aligned} &\mathbb{E}[\Upsilon_{\text{dB}}(x)] = H_x\hat{\theta} + \Psi^T(x)\Phi^{-1}(Y_m - H_m\hat{\theta}), \\ &\Sigma(x) = \hat{\alpha}^2 + \hat{\sigma}^2 - \Psi^T(x)\Phi^{-1}\Psi(x), \end{aligned} \tag{1}$$

where $Y_m = [y_1, ..., y_m]^T$ are the m priorly-collected CNR measurements (in dB), $X_m = [x_1, ..., x_m]$ are the measurement locations, $\hat{\theta}$, $\hat{\alpha}$, $\hat{\beta}$, and $\hat{\sigma}$ are the estimated channel parameters, $H_x = [1 \;\; -10\log_{10}(\|x - x_b\|)]$ with x_b denoting the location of the remote base station, $H_m = [H_{x_1}^T, ..., H_{x_m}^T]^T$, $\Psi(x) = [\hat{\alpha}^2\exp(-\|x - x_1\|/\hat{\beta}), ..., \hat{\alpha}^2\exp(-\|x - x_m\|/\hat{\beta})]^T$, and $\Phi = \Omega + \hat{\sigma}^2 I_m$ with $[\Omega]_{i,j} = \hat{\alpha}^2\exp(-\|x_i - x_j\|/\hat{\beta})$, $\forall i, j \in \{1, ..., m\}$ and I_m denoting the $m \times m$ identity matrix.

This model allows the robot to predict the CNR at any unvisited location, based on a few prior channel measurements which can be provided by static sensors in the field, gathered in previous operations or at the beginning of the operation, and/or obtained via crowdsourcing.[1]

For the case of fixed transmission power, this model allows us to find the probability that the robot is connected at a point x: $\text{Prob}(\Upsilon(x) \geq \Upsilon_{\text{th}}) = \mathbb{Q}((\Upsilon_{\text{th, dB}} - \mathbb{E}[\Upsilon_{\text{dB}}(x)])/\sqrt{\Sigma(x)})$, where $\mathbb{Q}(\cdot)$ represents the complementary cumulative distribution function of the standard normal distribution.

For the case of adaptive power, the expected required transmission power at location x is given by $\mathbb{E}[\Gamma_T(x)] = (2^r - 1)\mathbb{E}[1/\Upsilon(x)]/Z$, where $\mathbb{E}[1/\Upsilon(x)]$ can be evaluated based on the log-normal distribution of $\Upsilon(x)$, using the prediction parameters of (1), and $Z = -1.5/\ln(5p_{\text{BER}})$.

B. Motion Model

Based on experimental studies, a mobile robot's motion power can be modeled by a linear function of its speed for a large class of robots (e.g., a Pioneer robot) [12]: $\Gamma_m(v) = \kappa_1 v + \kappa_2$ with $0 < v \leq v_{\max}$, and $\Gamma_m(v) = 0$ for $v = 0$, where v and $v_{\max}$ are the robot's speed and maximum speed, respectively, and κ_1 and κ_2 are positive constants determined by the robot's mechanical system and the external load. When the robot travels at a constant speed $\tilde{v}$, the motion energy consumption for traversing a path P is given by $\mathscr{E}_m = (\kappa_1 + \kappa_2/\tilde{v})D(P)$, where $D(P)$ is the path length.

III. Problem Formulation

Consider the case where a robot needs to navigate from a starting position x_s to a given final position x_f in a 2D environment, $X \subset \mathbb{R}^2$, while avoiding obstacles. The obstacle region is denoted by $X_{\text{obs}} \subset X$, which can be an arbitrary set of locations. Let $P = [x_0, ..., x_K]$ represent the robot's path waypoints, where $x_i \in X$ is a waypoint $\forall i \in \{0, ..., K\}$, and the path is linearly interpolated between two consecutive waypoints. While moving from x_s to x_f, the robot needs to maintain communication with one or more remote stations, with Q denoting the set of these stations. To plan a path that satisfies the motion and communication requirements, we propose the following general optimization problem:

$$\begin{aligned} \min_{P} \quad & C(P) = C_c(P) + \delta C_m(P) \\ \text{s.t.} \quad & (1)\; x_0 = x_s,\; x_K = x_f, \quad (2)\; \mathscr{P}(P) \cap X_{\text{obs}} = \emptyset, \end{aligned} \tag{2}$$

where $C_c(P)$ and $C_m(P)$ are the respective communication and motion costs of the path P, and $\delta > 0$ weighs the importance of the two costs. The first constraint requires that the robot travels from the given start position to the given destination. The second constraint ensures that the robot does not run into obstacles, where $\mathscr{P}(P)$ is the path P with linear interpolation between two consecutive waypoints.

This general formulation encompasses a broad class of robotic communication tasks that are of interest to this paper, e.g., data uploading, broadcasting, and data relaying, as shown in Fig. 1. It is also applicable to both fixed and adaptive transmit power settings. Next, we discuss in detail the corresponding optimization problems for the different transmit power settings and communication tasks.

[1] The prior channel measurements that the robot can obtain are based on the downlink channel. When it needs to predict the uplink channel, the prior measurements can be collected by the remote station which can then send the needed information back to the robot for uplink channel prediction.

A. Fixed Transmission Power

First, consider the case where the robot cannot change its transmission power. As discussed in Sec. II-A, the robot's connectivity to the remote station depends on whether the CNR is above a prescribed threshold, Υ_{th}. Based on the channel prediction, we can calculate the probability that the CNR at location x is above the threshold: $\text{Prob}(\Upsilon(x) \geq \Upsilon_{\text{th}})$, to which we refer as the connection probability. To plan a path with good connectivity, we minimize the robot's travel in regions where the connection probability is below a given threshold, as shown in the following optimization:

$$\begin{aligned} \min_{P} \quad & D_{\text{nc}}(P) + \delta D(P) \\ \text{s.t.} \quad & (1)\ x_0 = x_s,\ x_K = x_f, \quad (2)\ \mathscr{P}(P) \cap X_{\text{obs}} = \emptyset, \end{aligned} \tag{3}$$

where $D_{\text{nc}}(P)$ is the length of the disconnected portion of the path and $\delta > 0$. The optimization effectively minimizes the disconnected portion of the path, while $D(p)$ prevents the robot from wandering excessively in a good-channel region without moving to the destination by including the total path length into the objective. In this way, the optimization formulation strikes a balance between communication and motion, with δ representing the corresponding weight.

Given a path $P = [x_0, ..., x_K]$, the disconnected length of the path is given as follows:

$$D_{\text{nc}}(p) = \sum_{i=1}^{k} \left(I_{\text{nc}}(x_{i-1}) \cdot \frac{\|x_i - x_{i-1}\|}{2} + I_{\text{nc}}(x_i) \cdot \frac{\|x_i - x_{i-1}\|}{2} \right), \tag{4}$$

where $\|x_i - x_j\|$ denotes the Euclidean distance between x_i and x_j. $I_{\text{nc}}(x)$ is a binary function indicating the robot's connectivity status at location x, with $I_{\text{nc}}(x) = 1$ indicating a disconnection and $I_{\text{nc}}(x) = 0$ denoting otherwise. Note that since the channel is predicted at discretized points (x_i's), we divide each segment of the path (i.e., the part between x_{i-1} and x_i) into two parts and use the connectivity prediction of the closer waypoint for each part, resulting in the two terms of (4). This is a good approximation of the disconnected length since the waypoints are typically densely selected.

The connectivity status indicator function, $I_{\text{nc}}(x)$, depends on the communication task, as we characterize next.

Uploading: Data uploading is possible as long as the robot is connected to one of the remote stations. Let $\Upsilon_q(x)$ denote the CNR associated with remote station q from location x. Then $I_{\text{nc}}(x)$ takes the following form:

$$I_{\text{nc}}(x) = \begin{cases} 0, & \exists q \in Q,\ \text{Prob}(\Upsilon_q(x) \geq \Upsilon_{\text{th}}) \geq p_{\text{th}}, \\ 1, & \text{otherwise}, \end{cases} \tag{5}$$

where p_{th} is a connection probability threshold. Thus, $I_{\text{nc}}(x) = 1$ when the robot does not have a good chance of connecting to any of the stations and $I_{\text{nc}}(x) = 0$ otherwise. The robot uses its probabilistic channel prediction over the workspace, as described in Sec. II-A, to find the distribution of $\Upsilon_q(x)$ over the workspace and characterize $I_{\text{nc}}(x)$.

Broadcasting: For broadcasting, the robot needs to transmit data to all remote stations, resulting in the following $I_{\text{nc}}(x)$:

$$I_{\text{nc}}(x) = \begin{cases} 0, & \forall q \in Q,\ \text{Prob}(\Upsilon_q(x) \geq \Upsilon_{\text{th}}) \geq p_{\text{th}}, \\ 1, & \text{otherwise}. \end{cases} \tag{6}$$

Relaying: In this case, the robot needs to maintain connection with both remote stations involved in the relay operation. As such, $I_{\text{nc}}(x)$ in this case takes the same form as in the broadcasting case, with the set Q representing the two remote stations. While there are 4 links involved in the relaying scenario, we are only concerned with the 2 uplinks from the robot to the stations, which are the energy-constrained ones.

B. Adaptive Transmission Power

We next consider the case where the robot can adapt its transmission power throughout its route. This allows the robot to ensure that the received SNR is always sufficient to satisfy the QoS requirements. Thus, if the robot has to transmit data at a location where the channel quality is poor, it will need to increase its transmission power. On the other hand, if the robot only focuses on minimizing its communication cost, it may spend too much motion energy finding regions with good channel quality. Therefore, our goal here is to design a path that minimizes the robot's total energy consumption in the operation area, as shown in the following optimization problem:

$$\begin{aligned} \min_{P} \quad & \mathscr{E}_c(P) + \delta \mathscr{E}_m(P) \\ \text{s.t.} \quad & (1)\ x_0 = x_s,\ x_K = x_f, \quad (2)\ \mathscr{P}(P) \cap X_{\text{obs}} = \emptyset, \end{aligned} \tag{7}$$

where $\mathscr{E}_c(P)$ and $\mathscr{E}_m(P)$ are the respective communication and motion energy costs of path P. Assume that the robot moves at a constant velocity $\tilde{v}$. Thus, as discussed in Sec. II-B, its motion energy becomes a linear function of the traveled distance. On the other hand, the communication energy of path P is given as follows:

$$\mathscr{E}_c(P) = \sum_{i=1}^{k} \left(\Gamma_c(x_i) \cdot \frac{\|x_i - x_{i-1}\|}{2\tilde{v}} + \Gamma_c(x_{i-1}) \cdot \frac{\|x_i - x_{i-1}\|}{2\tilde{v}} \right), \tag{8}$$

where $\Gamma_c(x)$ is the total power that is consumed for transmission at location x. At locations between two consecutive waypoints, the transmission power is taken to be the same as that of the closer waypoint.

Let $\Gamma_{T,q}(x)$ denote the required transmission power for the robot to transmit from location x to remote station q. The robot has a prediction of the CNR over the workspace (see Sec. II-A), based on which it can compute its expected required transmission power for location x as follows: $\mathbb{E}[\Gamma_{T,q}(x)] = (2^r - 1)\mathbb{E}[1/\Upsilon_q(x)]/Z$. The specific form of $\Gamma_c(x)$ depends on the communication task, as discussed next.

Uploading: For the uploading task, the robot uses the minimum required transmission power that allows for connection to a remote station: $\Gamma_c(x) = \min_{q \in Q}\ \mathbb{E}[\Gamma_{T,q}(x)]$.

Broadcasting: For broadcasting, the robot's transmission needs to reach all the remote stations. $\Gamma_c(x)$ is thus determined by the remote station that requires the highest transmission power: $\Gamma_c(x) = \max_{q \in Q}\ \mathbb{E}[\Gamma_{T,q}(x)]$.

Relaying: For relaying, the robot needs to simultaneously maintain communication with both remote stations (with indices 1 and 2) and relay the data between them, which results in: $\Gamma_c(x) = \mathbb{E}[\Gamma_{T,1}(x)] + \mathbb{E}[\Gamma_{T,2}(x)]$, where the summation indicates that there are two separate transmissions.

Overall, our proposed formulation encompasses various robotic communication operation scenarios, with different transmit power settings and communication tasks. Such path optimization problems are challenging to solve due to their high dimensionality, the presence of arbitrarily-shaped obstacles, and the spatially-varying wireless channel that cannot be captured analytically. The next section shows how to adapt RRT* to solve our path optimization with theoretical optimality guarantees and in the presence of obstacles.

IV. Path Planning for Robotic Communication

In this section, we show how to optimize the robot's path for communication operations by utilizing RRT* [10]. To properly apply RRT*, the cost function used in path planning (i.e., the objective function in (2)) must be additive, which provides the basis for the optimality of the path. We provide the formal definition of additivity as follows:

***Definition** 1 (Additivity): Given any two paths, P_1 and P_2, with the last point of P_1 being the first point of P_2 and their concatenation $P_1|P_2$, if $C(P_1|P_2) = C(P_1) + C(P_2)$, then the cost function C is additive.*

***Lemma** 1: The objective functions in problems (3) and (7) are additive, for all the three communication tasks.*

Proof: This can be easily confirmed given the definitions of the communication cost functions of (4) and (8), and the fact that the motion cost is a linear function of the traveled distance in both problems (3) and (7). ■

This ensures that we can use and adapt RRT* to find optimum paths for the robotic communication operations discussed in Sec. III. Furthermore, it should be noted that our proposed methodology is not limited to the cases of Sec. III. As long as the communication cost function can be put in an additive form, our proposed RRT*-based approach can be used. Next, we show how we adapt the RRT* to optimize the robot's path for communication operations.

A. Communication-Aware RRT*

RRT* is an efficient sampling-based path planning algorithm that iteratively builds a tree from a given source to a destination. As originally designed, the tree expansion and rewiring serve to reduce path length while avoiding obstacles. In this part, we show how to adapt RRT* to take into account costs of both motion and communication. Our approach is summarized in Alg. 1 and described in detail below.[2]

The algorithm incrementally builds a tree $G = (V, E)$ in the 2D space from x_s towards x_f, where V and E are the sets of nodes and edges, respectively. Each node represents a 2D location and an edge represents the path between two nodes. If $x_f \in V$, then there exists a feasible path in the tree from the root node x_s to x_f. We refer to the unique path from x_s to a node $x \in V \backslash \{x_s\}$ as a candidate partial path.

Sampling: In each iteration, the algorithm samples a random location, x_{rand}, and finds the nearest tree node to it, x_{nearest}, in terms of Euclidean distance. The algorithm then finds x_{new}, which is the closest point to x_{rand} within a prescribed radius, η, of x_{nearest}. If there are no obstacles blocking the direct path between x_{new} and x_{nearest}, then x_{new} will be added to V. Otherwise, the current x_{new} is discarded and the algorithm repeats this step until it obtains a feasible x_{new}.

Tree Expansion: Next, we need to find a proper parent for x_{new} such that the motion and communication costs are minimized. More specifically, consider a set of nodes, X_{near}, that have an obstacle-free direct path to x_{new} and are within a specified radius. The algorithm selects as the parent x_{min}, via which the robot will incur the minimum total cost when moving from x_s to x_{new}, i.e., $x_{\text{min}} = \text{argmin}_{x_{\text{near}} \in X_{\text{near}}} \tilde{C}(x_{\text{near}}) + C_m([x_{\text{near}}, x_{\text{new}}]) + C_c([x_{\text{near}}, x_{\text{new}}])$, where, for $x \in V$, $\tilde{C}(x)$ represents the full cost of the unique path in the tree from x_s to x, and $[x_{\text{near}}, x_{\text{new}}]$ is the direct path between the two corresponding nodes. This process ensures that the candidate partial path from x_s to x_{new} has the minimum total cost.

Rewiring: Given the updated tree with x_{new} added, some of the nodes can be rewired to reduce the costs of their corresponding candidate partial paths. For each node $x_{\text{near}} \in X_{\text{near}} \backslash \{x_{\text{min}}\}$, if the total cost of moving from x_s to x_{near} via the parent of x_{near} (i.e., $\tilde{C}(x_{\text{near}})$) is larger than that of going from x_s to x_{near} via x_{new}, i.e., $\tilde{C}(x_{\text{new}}) + C_m([x_{\text{new}}, x_{\text{near}}]) + C_c([x_{\text{new}}, x_{\text{near}}])$, then x_{near} is detached from its original parent node and added as a child to x_{new}. This rewiring results in continual path improvement and gives the algorithm the optimality property that we shall discuss in Sec. IV-B.

Final Path: The algorithm returns the minimum-cost path from x_s to x_f in the tree after running a prescribed number of iterations dictated by the computation budget. Note that a solution can be obtained from the algorithm any time after the first feasible path is found, which in our simulations generally takes around 1 s, depending on the obstacle configuration and the size of the space. After finding a first path, the algorithm continues to refine the quality of the solution as it runs more iterations. The final path from the tree is then converted to a sequence of densely and equally spaced waypoints that can be adapted for the size of the space.

As shown in the original RRT* paper [10], the algorithm drastically reduces the cost within the first few thousand iterations. Since RRT*'s computational complexity for executing n iterations is given by $\mathscr{O}(n\log(n))$ [13], these reductions happen quickly. Similarly, as we shall show in Sec. V, our communication-aware RRT* performs the majority of cost reduction within an initial short period of time.

B. Theoretical Analysis

In this part, we study the optimality and the theoretical properties of our communication-aware RRT*. First, we show the asymptotic optimality of our proposed approach.

***Theorem** 1: For the robotic communication scenarios in Sec. III, the cost of the path given by Alg. 1 converges to the optimum almost surely, i.e., $Prob(lim_{n\to\infty}C(P_{n,min}) = C(P^\star)) = 1$, where n is the number of iterations, $P_{n,min}$ is the minimum-cost path in the tree from x_s to x_f after n iterations, and $P^\star$ is the optimum path. The path given by Alg. 1 also satisfies the optimization constraints and avoids obstacles.*

Proof: As shown in Lemma 1, the cost function C is additive for all the transmission power settings and communication tasks discussed in Sec. III. Therefore,

[2] See [10] for an in-depth description of the original RRT*.

Algorithm 1: Communication-Aware RRT*

$V = \{x_s\}$, $E = \emptyset$
while *within computation budget* **do**
  $G = (V, E)$
  $x_{\text{rand}} = Sample(X, X_{\text{obs}})$
  $x_{\text{nearest}} = Nearest(G, x_{\text{rand}})$
  $x_{\text{new}} = \text{argmin}_{\|x - x_{\text{nearest}}\| \leq \eta} \|x - x_{\text{rand}}\|$
  if $ObstacleFree(x_{new}, x_{nearest})$ **then**
    `// Tree Expansion`
    $V = V \cup x_{\text{new}}$
    $X_{\text{near}} = Near(G, x_{\text{new}}, X, X_{\text{obs}})$
    $x_{\text{min}} = \underset{x_{\text{near}} \in X_{\text{near}}}{\text{argmin}} \tilde{C}(x_{\text{near}}) + C_m([x_{\text{near}}, x_{\text{new}}]) + C_c([x_{\text{near}}, x_{\text{new}}])$
    $E = E \cup \{(x_{\text{min}}, x_{\text{new}})\}$
    `// Rewiring`
    for $x_{near} \in X_{near} \setminus \{x_{min}\}$ **do**
      $x_{\text{parent}} = Parent(x_{\text{near}})$
      if $\tilde{C}(x_{near}) > \tilde{C}(x_{new}) + C_m([x_{new}, x_{near}]) + C_c([x_{new}, x_{near}])$ **then**
        $E = E \setminus \{(x_{\text{parent}}, x_{\text{near}})\}$
        $E = E \cup \{(x_{\text{new}}, x_{\text{near}})\}$
      end
    end
  end
end
Return minimum-cost path from x_s to x_f in G

the communication-aware RRT* is asymptotically optimal, based on Theorem 21 of [10]. ■

Next, we study the properties of the optimum path, $P^\star$, in the adaptive transmit power setting and compare it with the shortest obstacle-free path, P_s. We are interested in characterizing $\Delta D = D(P^\star) - D(P_s)$, the additional incurred path distance due to considering communication costs. Let $\Gamma_m(\tilde{v})$ denote the motion power for a constant speed of $\tilde{v}$, as defined in Sec. II-B. Furthermore, $\bar{\Gamma}_c^s = \frac{1}{K}\sum_{x \in P_s} \Gamma_c(x)$ is the average transmit power over the waypoints of P_s and $\bar{\Gamma}_c^\star = \frac{1}{K}\sum_{x \in P^\star} \Gamma_c(x)$ is the average transmit power over the waypoints of $P^\star$. We assume that the robot adapts its power at equally-spaced points along the path (e.g., at the waypoints). The following result characterizes the relationship between ΔD and the motion and communication costs.

***Proposition** 1: The following inequality holds:*

$$\Delta D \leq D(P_s)(\bar{\Gamma}_c^s - \bar{\Gamma}_c^\star)/(\delta \Gamma_m(\bar{v}) + \bar{\Gamma}_c^\star). \tag{9}$$

Proof: Since $P^\star$ should incur less total cost than P_s, we have $\frac{D(P_s)}{\bar{v}}(\delta\Gamma_m(\bar{v}) + \bar{\Gamma}_c^s) \geq \frac{D(P_s)+\Delta D}{\bar{v}}(\delta\Gamma_m(\bar{v}) + \bar{\Gamma}_c^\star)$. By rearranging the terms, the inequality can be obtained. ■

It can be seen that the extent of deviation from the shortest path depends on the amount of communication savings in the optimum path, as well as the robot's motion power. If the average transmission power along P_s goes up, then greater communication energy is saved by taking $P^\star$, which can make up for the additional energy spent for traveling a longer path. On the other hand, if the robot's motion power goes up, then savings in communication energy may not justify a larger additional distance, so ΔD will become smaller.

V. Simulation Experiments

In this section, we solve the path optimization problems using our proposed approach in realistic simulated 2D wireless channel environments and in the presence of obstacles.

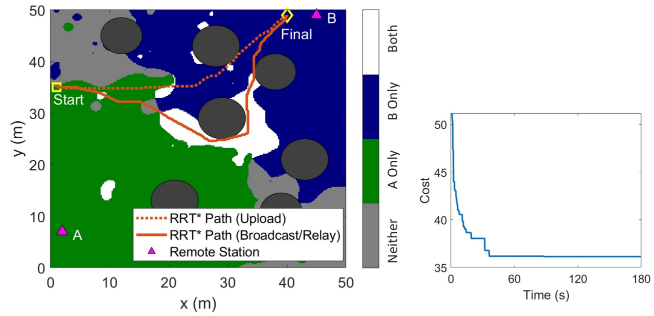

Fig. 2: (Left) Paths from solving (3) with our proposed approach for the case of two remote stations. The orange solid (dashed) curve shows the path found for the upload (broadcast/relay) task. The green (blue) regions are where the robot predicts it can only connect to remote station A (B). The white (grey) color indicates that the robot predicts it can connect to both (neither) of the stations. (Right) Reduction of cost (i.e., objective of (3)) as a function of Alg. 1's run time in the case of broadcast/relay. See the color pdf to better view this figure.

We consider a 50 m×50 m discretized 2D workspace consisting of 0.2 m×0.2 m regular grids, with randomly-sized, randomly-placed circular obstacles and two remote stations. The wireless channel for each remote station is simulated by using [14], with the following channel parameters: $\hat{\theta} = [-41.34, 3.86]^T$, $\hat{\alpha} = 10.24$, $\hat{\beta} = 3.09$ m, and $\hat{\sigma} = 3.2$, which are obtained from real channel measurements [11]. The receiver noise power is -100 dBm. The robot predicts the channel based on 0.8% prior channel samples randomly located in the space, using the prediction framework of Sec. II-A. The communication-aware RRT* of Alg. 1 is run for 3 minutes to obtain the path. We further compare with the benchmark of using the original RRT*.

A. Case of Fixed Transmission Power

We first consider the fixed power setting. The SNR threshold is 20 dB and the transmit power is 120 mW. The connection probability threshold is $p_{\text{th}} = 0.7$. We choose $\delta = 0.01$ to find highly connected path.

The robot navigates from $(1, 35)$ to $(40, 49)$. There are two remote stations in the field at $(2, 7)$ and $(45, 49)$. Fig. 2 (left) shows the paths computed by our proposed approach for the upload and the broadcast/relay tasks (broadcast and relay have the same cost function in this fixed transmission power setting, resulting in the same path). It can be seen that for the upload task, the robot mostly travels in regions that are colored in either blue or green, since it only needs to connect to one of the two remote stations. This allows it to take a more direct path towards the destination while staying connected the entire time. As for broadcast/relay, the robot needs to maintain connection to both stations, which is a stricter requirement. As such, the robot takes a much larger detour to travel as much as possible in the white-colored regions, where it can connect to both remote stations.

While we run Alg. 1 for 3 minutes, it achieves most of the cost reduction quickly, as shown in Fig. 2 (right). For instance, it achieves 87% of all cost reduction within the first 30 s, reducing the cost from 51.13 m to 38.04 m. This demonstrates the computational efficiency of our approach.

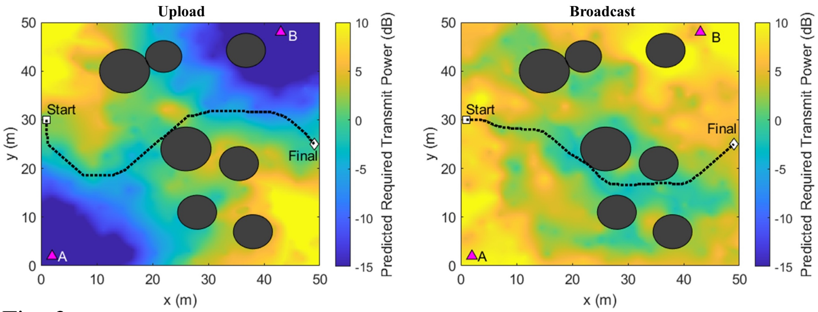

Fig. 3: Paths from solving (7) by using our proposed approach for upload (left) and broadcast (right) cases. In each figure, the orange dashed curve shows the path given by Alg. 1. The white square (diamond) indicates the start (final) position. The magenta triangles indicate the locations of the two remote stations A and B. The disks are the obstacles. A brighter (darker) color in the colormap indicates a higher (lower) predicted required transmit power. See the color pdf to better view this figure.

Scenarios	Upload	Broadcast	Relay
Fixed Pwr. *(Disc. Red.)*	67%	25%	25%
Adaptive Pwr. *(Energy Red.)*	14%	16%	13%

TABLE I: Performance improvement given by our proposed approach over the benchmark of using the original RRT*. We use the reduction of the disconnected portion of the path and the reduction of the total energy cost as the evaluation metrics for the fixed and adaptive transmit power settings, respectively.

Next, we show the performance (i.e., communication cost) as the robot travels its designed path and measures the true value of the channel. We further run 20 channel realizations and report the average performance. We also compare with the original-RRT* benchmark which only considers motion costs. For all tasks, Table I shows that our approach significantly reduces the percentage of the path that is disconnected as compared to the benchmark.

B. Case of Adaptive Transmission Power

Next, consider the case of adaptive power. In these experiments, the robot moves from $(1,30)$ to $(49,25)$. There are two remote stations at $(2,2)$ and $(43,48)$. The required BER is 10^{-6}, and the required spectral efficiency is 8 bits/s/Hz for upload and 6 bits/s/Hz for broadcast/relay. The motion parameters are $\kappa_1 = 7.4\,\text{N}$ and $\kappa_2 = 0.29\,\text{W}$, based on real-world measurements of a Pioneer robot [12], and $\tilde{v} = 1\,\text{m/s}$. Prioritizing communication energy, we set $\delta = 0.1$.

Fig. 3 shows sample paths obtained by using Alg. 1 to solve (7) for the tasks of upload and broadcast. In Fig. 3 (left), the robot performs data uploading and only needs to transmit to one of the remote stations at any point during the trip. As such, it first moves near station A and later near station B in the path. Fig. 3 (right) shows the resulting path for broadcast. Since the robot's transmission must cover both remote stations, it transmits with the maximum of the two stations' required transmit powers. As such, the robot takes a path that keeps it roughly equidistant to the two stations. In addition, it exploits the local good-channel areas (e.g., by hugging the side of the obstacle centered at $(26,24)$). As for the relay case, the path is similar to that of the broadcast scenario and is thus omitted for brevity.

We next average the true communication and motion energy cost as the robot travels its designed path over 20 channel realizations. When comparing with the benchmark, our proposed approach incurs considerably less total energy cost, reducing it by 14%, 16%, and 13% on average for upload, broadcast, and relay, respectively, as shown in Table I.

Overall, these results show that our proposed approach generates paths with good communication qualities (i.e., small disconnected portions or low energy costs) for various robotic communication operations, while avoiding obstacles.

VI. Conclusions

In this paper, we considered a robot that needs to perform a given communication task (e.g., data uploading, broadcasting, or relaying) while navigating from a start position to a designated final position, avoiding obstacles, and minimizing its total motion-communication cost. We showed how we can adapt the traditional RRT* path planning algorithm to jointly consider both communication and motion in realistic channel environments that contain obstacles. We further mathematically showed the optimality of our proposed algorithm and characterized properties of the optimum path. Finally, by extensive evaluations in realistic channel environments, we showed that our approach can produce paths with considerably better connectivity or much lower total energy costs as compared to the benchmark.

References

[1] Q. Wu et al. Joint trajectory and communication design for multi-uav enabled wireless networks. *IEEE Trans. on Wireless Commun.*, 2018.
[2] A. Ghaffarkhah and Y. Mostofi. Channel learning and communication-aware motion planning in mobile networks. In *Proceedings of the American Control Conference*, 2010.
[3] A. Muralidharan and Y. Mostofi. Communication-aware robotics: Exploiting motion for communication. *Annual Reviews of Control, Robotics, and Autonomous Systems*, 2020.
[4] J. Fink, A. Ribeiro, and V. Kumar. Robust control of mobility and communications in autonomous robot teams. *IEEE Access*, 2013.
[5] Y. Wu et al. Communication-motion planning for wireless relay-assisted multi-robot system. *IEEE Wireless Commun. Lett.*, 2016.
[6] U. Ali, H. Cai, Y. Mostofi, and Y. Wardi. Motion-communication co-optimization with cooperative load transfer in mobile robotics: An optimal control perspective. *IEEE Trans. Control Netw. Syst.*, 2019.
[7] Y. Liu et al. Joint optimization of path planning and resource allocation in mobile edge computing. *IEEE Trans. on Mobile Comput.*, 2019.
[8] A. Ghaffarkhah and Y. Mostofi. Dynamic networked coverage of time-Varying environments in the presence of fading communication channels. *ACM Trans. Sens. Netw.*, 2014.
[9] A. Muralidharan and Y. Mostofi. Path planning for minimizing the expected cost until success. *IEEE Trans. on Robot.*, 2019.
[10] S. Karaman and E. Frazzoli. Incremental sampling-based algorithms for optimal motion planning. *Robotics: Science and Systems*, 2010.
[11] M. Malmirchegini and Y. Mostofi. On the spatial predictability of communication channels. *IEEE Trans. on Wireless Commun.*, 2012.
[12] Y. Mei, Y. Lu, Y. Hu, and C. Lee. Deployment of mobile robots with energy and timing constraints. *IEEE Trans. on Robot.*, 2006.
[13] S. Karaman and E. Frazzoli. Sampling-based algorithms for optimal motion planning. *Int. J. Robot. Res.*, 2011.
[14] A. Gonzalez-Ruiz, A. Ghaffarkhah, and Y. Mostofi. A comprehensive overview and characterization of wireless channels for networked robotic and control systems. *J. Robot.*, 2011.