Motion and Communication Co-optimization with Path Planning and Online Channel Prediction[†]

Usman Ali[*], Hong Cai[**], Yasamin Mostofi[**] and Yorai Wardi[*]

***Abstract*— This paper considers the problem of optimally balancing motion energy and communication transmission energy of a mobile robot tasked with transmitting a given number of data bits to a remote station, while navigating to a pre-specified destination in a given amount of time. The problem is cast in the setting of optimal control, where the robot has to choose its path, acceleration, and transmission rate along the path so as to minimize its energy required for transmission and motion, while satisfying various power and communication constraints. We use realistic models for the robot's channel estimation, motion dynamics, and power and energy costs. The main contribution of the paper is to show how to co-optimize robot's path along with other communication and motion variables. Two versions of the problem are solved: the first is defined offline by assuming that all the channel measurements are taken before the robots starts moving, while in the second the channel estimation is updated while the robot is in motion, and hence it is solved online. In both cases we utilize an in-house algorithm that computes near-optimal solutions in little time, which enables its use in the online setting. The optimization strategy is described in detail and validated by simulation of realistic scenarios.**

I. Introduction

Communication-aware mobile robotics is an emergent field of enquiry whose origins are in two related areas that have been extensively researched in the past twenty years: mobile sensor networks [1]–[3], and networked robotics systems [4]–[8]. A central problem in communication-aware robotics is the co-optimization of sensing, communication, and navigation under physical and resource constraints [9]–[13]. More specifically, the problem of balancing transmission energy with motion energy, in realistic communication environments, has been the focus of research in recent years [11], [12], [14], [15], and it is also the subject of this paper.

While optimization of transmission energy and motion energy has been traditionally explored separately in the respective literatures on communications and robotics (e.g., [16]–[18]), only recently the problem of co-optimizing the two forms of energy has begun to attract attention. In [19], the authors propose an efficient approximate path planning algorithm that minimizes motion and communication energy costs. Ref. [20] optimizes relay configurations in data-intensive wireless sensor networks. In [21], the authors develop an algorithm for maximizing the lifetime of wireless sensor networks considering both communication and motion costs of the sensors. Ref. [22] considers a dynamic co-optimization problem in the setting of optimal control and develops a Hamiltonian-based algorithm for its solution. All of the references [19]–[22] focus on the robotics and optimization aspects of the problem while using over-simplified models for channel and communication energy costs. Realistic models of channel fading [12] are used in [14], [15] in designing a co-optimization strategy for balancing a robot's speed with transmission rate, and [23] develops an effective algorithm for realizing that strategy.

Ref. [23] is the starting point of this paper. It considers a robot required to transmit a given number of bits in a given amount of time to a remote station, while traversing a predetermined path. The channel quality is variable along the path, and it is predicted by the realistic model described in [14]. The considered problem is to compute the profiles of the robot's acceleration and spectral efficiency (transmission rate per unit bandwidth) that minimize the combined energy spent on transmission and motion. We cast the problem in the framework of optimal control, and solved it by using an in-house algorithm that is simple to code and was shown to yield fast convergence.

This paper extends the problem and methodology developed in [23] in the following two ways. First, it adds the challenging element of path planning by lifting the restriction that the robot has to follow a pre-determined path. Rather, it has to compute an optimal trajectory. Second, channel prediction is updated as the robot gathers more channel samples in the workspace, requiring a re-evaluation of the optimal trajectory for the cost to go, in contrast to [23] which does a one-time channel prediction and co-optimization before the robot starts moving. We point out that we do not use model-predictive control or rolling horizons, but rather compute the entire trajectory of the cost-to-go performance functional to the given final time. These enhancements over the setting in [23] pose significant computational challenges. The proposed scheme handles these challenges in an effective way, as shown by satisfactory solution for the example-problems presented in the sequel.

The rest of the paper is organized as follows. Section II formulates the problem and discusses relevant existing results. Section III, considering an elaborate example, solves the problem offline, while Section IV applies the online version of the algorithm. Finally, Section V concludes the paper and points out various directions for future research.

[†]Research supported in part by the NSF under Grant Numbers CNS-1239225 and NeTS-1321171.

[*]School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332. Email: usmanali@gatech.edu, ywardi@ece.gatech.edu.

[**]Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106. Email: hcai@ece.ucsb.edu, ymostofi@ece.ucsb.edu.

II. Problem Formulation

This section formulates the optimal control problem, describes the channel prediction technique employed, presents the algorithm used, and recounts results of its application to the fixed-path problem presented in [23].

A. Problem Definition.

Consider a robot that has to traverse a path between a source point $S \in \mathscr{R}^2$ and a destination point $D \in \mathscr{R}^2$ while transmitting a given number of bits to a remote station in a given time-horizon $[0,t_f]$. The problem is to determine the robot's path, acceleration, and transmission rate as functions of time $t \in [0,t_f]$ so as to minimize the total energy required for transmission and motion. Based on few measurements, the channel quality is predicted probabilistically as will be detailed below. The power required for motion depends on the robot's velocity and acceleration, while its transmission power depends on its position relative to the remote station, transmission rate, and the channel quality (shadowing and multipath fading). For the robot's motion, we use the second order dynamical model

$$\begin{aligned} \dot{x}_1(t) &= x_2(t), \\ \dot{x}_2(t) &= u(t), \end{aligned} \tag{1}$$

where $x_1 \in \mathscr{R}^2$ is the position of the robot in the plane, $x_2 \in \mathscr{R}^2$ denotes its velocity, and $u \in \mathscr{R}^2$ is its acceleration. The initial condition of this equation is $x_1(0) = S$ and $x_2(0) = 0$. According to Ref. [17], the power required for the robot's motion has the form, for given constants $k_i \geq 0$, $i = 1,\ldots,6$,

$$\begin{aligned} P_m(t) = k_1||u(t)||^2 + k_2||x_2(t)||^2 + k_3||x_2(t)|| + k_4 \\ + k_5||u(t)|| + k_6||u(t)|| \cdot ||x_2(t)||, \end{aligned} \tag{2}$$

The power required for transmitting data to the remote station (from $x_1(t)$) is given by

$$P_c(t) = \frac{2^{R(t)} - 1}{K} s(x_1(t)), \tag{3}$$

where $R(t) \geq 0$ is the spectral efficiency of the channel at time t and position $x_1(t)$, K is a constant depending on the threshold bit error rate acceptable at the receiver, and $s(x_1(t))$ is the estimated channel quality metric at position $x_1(t) \in \mathscr{R}^2$; see Section II-B for details. Let Q be the total number of bits the robot has to transmit, then the requirement of transmitting Q bits in time interval $[0,t_f]$ results in the constraint

$$\int_0^{t_f} R(t)dt = \frac{Q}{B} := c,$$

where B denotes the channel's bandwidth. To get rid of the integral, so as to make this constraint more amenable to our algorithm, we introduce an auxiliary state variable, $x_3 \in \mathscr{R}$, defined by the equation

$$\dot{x}_3 = R(t), \tag{4}$$

with the boundary conditions $x_3(0) = 0$ and $x_3(t_f) = c$. Other final-time constraints on the state variable (position and velocity) are $x_1(t_f) = D$ and $x_2(t_f) = 0$. We also assume upper-bound constraints on $u(t)$ and $R(t)$ of the form

$$0 \leq ||u(t)|| \leq u_{\max}, \quad 0 \leq R(t) \leq R_{\max}, \tag{5}$$

for given $u_{\max} > 0$ and $R_{\max} > 0$.

The related optimal control problem is defined as follows. Its input is $(u(t),R(t)) \in \mathscr{R}^2 \times \mathscr{R}$, $t \in [0,t_f]$, its state is $(x_1(t),x_2(t),x_3(t))$, and its dynamics are given by Eqs. (1) and (4) with the initial conditions $x_1(0) = S$, $x_2(0) = 0$, and $x_3(0) = 0$. Its performance function, to be minimized, is

$$\bar{J} := \int_0^{t_f} \big(P_m(t) + \gamma P_c(t)\big)dt, \tag{6}$$

where $P_m(t)$ and $P_c(t)$ are the motion power and transmission power defined, respectively, by Eqs. (2) and (3), and $\gamma > 0$ is a given constant. The problem is to minimize $\bar{J}$ subject to the above dynamic equations, the upper-bound constraints on the input as defined by Eq. (5), and the final-state constraints $x_1(t_f) = D$, $x_2(t_f) = 0$, and $x_3(t_f) = c$.

We handle the final-state constraints with a penalty function of the form $C_1||x_1(t_f) - D||^2 + C_2||x_2(t_f)||^2 + C_3||x_3 - c||^2$, for constants $C_1 > 0$, $C_2 > 0$, and $C_3 > 0$, chosen large enough so that the terminal constraints are almost satisfied. The resulting optimal control problem now has the following form: Minimize the cost functional J defined as

$$\begin{aligned} J = \int_0^{t_f} \Bigg(\frac{2^{R(t)} - 1}{K} s(x_1) + \gamma\big(k_1||u(t)||^2 + k_2||x_2(t)||^2 \\ + k_3||x_2(t)|| + k_4 + k_5||u(t)|| + k_6||u(t)||||x_2(t)||\big)\Bigg)dt \\ + C_1||x_1(t_f) - D||^2 + C_2||x_2(t_f)||^2 + C_3||x_3(t_f) - c||^2, \end{aligned} \tag{7}$$

subject to the dynamic equations

$$\begin{aligned} \dot{x}_1(t) &= x_2(t), & x_1(0) &= S \\ \dot{x}_2(t) &= u(t), & x_2(0) &= 0 \\ \dot{x}_3(t) &= R(t), & x_3(0) &= 0, \end{aligned}$$

and constraints on the control inputs given by (5).

B. Online Channel Prediction

In order for the robot to plan its trajectory and communication/motion parameters, it needs an assessment of the link quality in communication from unvisited locations to the remote node. In this section, we briefly summarize our past work on channel prediction, which we shall utilize in the rest of the paper. Assuming the common MQAM modulation for a robot's communication to the remote station, the required transmit power at time t can be characterized as in [24] by

$$\tilde{P}_c(t) = (2^{R(t)} - 1)/(K\Upsilon(x_1(t))), \tag{8}$$

where $K = -1.5/\ln(5p_{b,\text{th}})$, $p_{b,\text{th}}$ is the given Bit Error Rate (BER) threshold at the receiver, $R(t)$ is the spectral efficiency at time t and $\Upsilon(x_1(t))$ the instantaneous channel-to-noise ratio (CNR) at $x_1(t)$. It is well known that the CNR can be modeled as a random process with three components: path loss, shadowing and multipath fading [24]. As shown in [12], based on a small number of a priori channel measurements, a Gaussian random variable, $\Upsilon_{\text{dB}}(q)$, can best characterize the CNR (in the dB domain) at an unvisited location q, the mean and variance of which are given by

$$\begin{aligned} \overline{\Upsilon}_{\text{dB}}(q) &= H_q\hat{\theta} + \Psi^{\text{T}}(q)\Phi^{-1}\big(Y - H_{\mathscr{Q}}\hat{\theta}\big), \\ \Sigma(q) &= \hat{\xi}^2_{\text{dB}} + \hat{\rho}^2_{\text{dB}} - \Psi^{\text{T}}(q)\Phi^{-1}\Psi(q), \end{aligned}$$

where Y is the stacked vector of m a priori gathered CNR measurements, $\mathscr{Q} = \{q_1,...,q_m\}$ denotes the measurement positions, $H_q = [1 \ -10\log_{10}(\|q - q_b\|)]$, $H_{\mathscr{Q}} =$

$[H_{q_1}^T \ ... \ H_{q_m}^T]^T$, $\Phi = \Omega + \hat{\rho}_{\rm dB}^2 I_m$ with $[\Omega]_{i,j} = \hat{\xi}_{\rm dB}^2 \exp(-\|q_i - q_j\|/\hat{\eta})$, for $i,j \in \{1,...,m\}$, and $\Psi(q) = [\hat{\xi}_{\rm dB}^2 \exp(-\|q - q_1\|/\hat{\eta}) \ ... \ \hat{\xi}_{\rm dB}^2 \exp(-\|q - q_m\|/\hat{\eta})]^T$. The terms $\hat{\theta} = [\hat{K}_{\rm PL} \ \hat{n}_{\rm PL}]^T$, $\hat{\xi}_{\rm dB}$, $\hat{\eta}$ and $\hat{\rho}_{\rm dB}$ are the estimated channel parameters. See [12] for more details on the estimation of channel parameters and the performance of this framework in channel prediction. Based on this framework, the CNR at unvisited location $x_1(t)$ can be predicted as a lognormal random variable (in the linear domain). The expected transmit power $P_{\rm c}(t)$ is given by

$$P_{\rm c}(t) = \frac{2^{R(t)} - 1}{K} E\left[\frac{1}{\Upsilon(x_1(t))}\right]. \tag{9}$$

Note that for lognormally distributed $\Upsilon(x_1(t))$, we have

$$E\left[\frac{1}{\Upsilon(x_1(t))}\right] = \exp\left(\left(\frac{\ln 10}{10}\right)^2 \frac{\Sigma(x_1(t))}{2}\right) \frac{1}{\overline{\Upsilon}(x_1(t))}, \tag{10}$$

where $\overline{\Upsilon}(x_1(t)) = 10^{\overline{\Upsilon}_{\rm dB}(x_1(t))/10}$. Equation (10) provides an estimate of the predicted channel quality at $x_1(t)$ and we let $s(x_1(t)) = E[1/\Upsilon(x_1(t))]$, substituting which in Eq. (9) leads to Eq. (3) for computing the transmit power.

Based on few channel measurements collected a priori (e.g. by static sensors in the field), an initial prediction of channel quality over the workspace can be performed and the optimal control problem (7) solved for optimal controls. As the robot moves, it gathers additional channel measurements (e.g. by gathering more samples along its path, through crowdsourcing and/or by other robots in the field), which enables it to predict the channel quality more accurately. Thus the channel prediction and subsequent optimization is performed from time to time over the remaining time horizon $[t_0, t_f]$, where t_0 denotes the present time at which the optimization is performed. The details of this online optimization procedure will be presented in Section IV.

C. Hamiltonian-Based Algorithm

Recently we developed an algorithm which is suitable for a class of power-aware optimal control problems [22], [23], [25]. Cumulative experience with it reveals some favorable computational properties including fast convergence towards a local minimum. This does not mean fast asymptotic convergence, which characterizes an algorithm's behavior close to a local minimum, but rather large strides towards a region of a (local) minimum. A key innovation in the algorithm is its choice of a descent direction, which is not based on gradient descent but rather follows an alternative approach requiring little computing efforts. We next explain the structure of the algorithm and summarize its performance on the power-aware problem considered in [23].

Consider the abstract Bolza optimal control problem where the system's dynamics are defined by the equation:

$$\dot{x} = f(x,u), \quad x(0) := x_0,$$

where $x \in \mathscr{R}^n$, $u \in \mathscr{R}^k$, and $f : \mathscr{R}^n \times \mathscr{R}^k \to \mathscr{R}^n$ is Lipschitz continuous in x and continuous in u. Given a final time $t_f > 0$, a cost function $L : \mathscr{R}^n \times \mathscr{R}^k \to \mathscr{R}$, and a terminal-state cost function $\phi : \mathscr{R}^n \to \mathscr{R}$, define the cost functional as

$$J := \int_0^{t_f} L(x,u)dt + \phi(x(t_f)).$$

The optimal control problem, considered, is to minimize J subject to the pointwise constraints $u(t) \in \mathscr{U}$, where $\mathscr{U} \subset \mathscr{R}^k$ is an input-constraint set. We make the following assumption:

Assumption 1: 1). The function $f(x,u)$ is affine in $u \in \mathscr{U}$ for every $x \in \mathscr{R}^n$, and the function $L(x,u)$ is convex in $u \in \mathscr{U}$ for every $x \in \mathscr{R}^n$.

2). The set $\mathscr{U}$ is compact and convex.

Let $p(t)$, $t \in [0,t_f]$, denote the costate (adjoint) trajectory defined by the equation

$$\dot{p} = -\Big(\frac{\partial f}{\partial x}(x,u)\Big)^\top p - \Big(\frac{\partial L}{\partial x}\Big)^\top$$

with the boundary condition $p(t_f) = \nabla\phi(x(t_f))$, and let

$$H(x,u,p) := p^\top f(x,u) + L(x,u)$$

denote the Hamiltonian function (see, e.g., Ref. [26]). The kind of problems for which our algorithm is suitable have the property that, for given $x \in \mathscr{R}^n$ and $p \in \mathscr{R}^n$, a minimum value of the Hamiltonian $H(x,w,p)$, over $w \in \mathscr{U}$, can be computed via a simple, explicit formula. Next we describe the algorithm where we refer to the control function $u(t)$, $t \in [0,t_f]$, by the boldface notation $\mathbf{u}$.

ALGORITHM

Parameters: Constants $\alpha \in (0,1)$ and $\beta \in (0,1)$. Given a control $\mathbf{u}$, compute the next control $\mathbf{u}_{next}$ as follows:

1. Direction from $\mathbf{u}$*:* Compute the state and costate trajectories $x(t)$ and $p(t)$, $t \in [0,t_f]$. For every $t \in [0,t_f]$, compute a pointwise (t-dependent) minimizer of the Hamiltonian denoted by $u^\star(t)$, namely, a point $u^\star(t) \in \mathscr{U}$ satisfying

$$u^\star(t) \in \operatorname{argmin}\Big(H(x(t),w,p(t)) \mid w \in \mathscr{U}\Big).$$

Define $\mathbf{u}^\star$ to be the function $u^\star(t)$, $t \in [0,t_f]$.[1] Define the direction from $\mathbf{u}$ to be $d(t) := u^\star(t) - u(t)$, namely, in functional notation, $\mathbf{d} = \mathbf{u}^\star - \mathbf{u}$.

2. Step size along the direction $\mathbf{d}$*:* Define

$$\theta(\mathbf{u}) = \int_0^{t_f} \big(H(x(t),u^\star(t),p(t)) - H(x(t),u(t),p(t))\big)dt.$$

Compute $k \in \{0,1,\dots,\}$ defined as

$$k = \min\Big\{j = 0,1,\dots \Big| J(\mathbf{u} + \beta^j \mathbf{d}) - J(\mathbf{u}) \le \alpha\beta^j\theta(\mathbf{u})\Big\}, \tag{11}$$

and set the step size, λ, to be $\lambda = \beta^k$.

3. Update: Set $\mathbf{u}_{next}$ to be

$$\mathbf{u}_{next} = \mathbf{u} + \lambda\mathbf{d}.$$

As we pointed out, the main innovation of the algorithm is in the choice of the direction in Step 1, while Step 2 describes a standard Armijo step size; see [27] for extensive discussions thereof. The direction is not based on explicit gradient computations but rather comprises a form of conjugate gradient, and we believe that this plays a role in the fast convergence of the algorithm that has been noted in various simulation experiments. In particular, Fig. 2 and Table I in [23], considering a one-dimensional power-aware problem, show that the algorithm makes most of its descent towards the minimum value of J in under 10 iterations. The next section exhibits similar performance of the algorithm for the path-planning problem at hand.

[1] There may arise measurability issues due to the explicit characterization of $u^\star(t)$ for all t in the uncountable set $[0,t_f]$. However, in a grid-based implementation these issues will be avoided since t would lie in a finite set.

III. Path Planning with Motion and Communication Co-optimization

In this section, we consider the application of the algorithm to the problem defined in section II. The Hamiltonian associated with the optimal control problem (7) is

$$H(x,[u,R],p) = p_1^T x_2 + p_2^T u + p_3 R + \frac{2^R-1}{K} s(x_1) + \gamma\big(k_1||u||^2 + k_2||x_2||^2 + k_3||x_2|| + k_4 + k_5||u|| + k_6||u||||x_2||\big), \quad (12)$$

where the costates $p_1 \in \mathscr{R}^2$, $p_2 \in \mathscr{R}^2$, and $p_3 \in \mathscr{R}$ are defined by the adjoint equations

$$\dot{p}_1 = -\frac{2^R-1}{K}\frac{\partial s(x_1)}{\partial x_1},$$
$$\dot{p}_2 = -p_1 - \gamma\left(2k_2x_2 + k_3\frac{x_2}{||x_2||} + k_6||u||\frac{x_2}{||x_2||}\right),$$
$$\dot{p}_3 = 0,$$

with terminal constraints $p_1(t_f) = 2C_1(x_1(t_f) - D)$, $p_2(t_f) = 2C_2x_2(t_f)$ and $p_3(t_f) = 2C_3(x_3(t_f) - c)$, respectively.

In the forthcoming we assume that $k_5 = k_6 = 0$ in (12) as we did in [17]. The minimizer of this Hamiltonian subject to the input constraints can be seen to be given by

$$u^\star = \begin{cases} -\dfrac{p_2}{2\gamma k_1}, & \text{if } \dfrac{1}{2\gamma k_1}||p_2|| \le u_{\max} \\ -\dfrac{p_2}{||p_2||}u_{max}, & \text{if } \dfrac{1}{2\gamma k_1}||p_2|| > u_{\max}, \end{cases}$$

$$R^\star = \begin{cases} \dfrac{1}{\ln(2)}\ln\left(\dfrac{-p_3K}{\ln(2)s(x_1)}\right), & \text{if } p_3 \le -\dfrac{(\ln(2)s(x_1))}{K} \\ R_{\max}, & \text{if } \dfrac{1}{\ln(2)}\ln\left(\dfrac{-p_3K}{\ln(2)s(x_1)}\right) > R_{\max} \\ 0, & \text{otherwise.} \end{cases}$$

Application:

Consider a robot that is tasked to move from the initial point $S = (20,40)$ to the final point $D = (10,5)$ in the plane, and it has to transmit 150 bits/Hz to a remote station located at $q_b = (5,5)$ in 40 seconds. The acceleration and spectral efficiency can take maximum values of $u_{max} = 0.5m/s^2$ and $R_{max} = 6 Bits/Hz/sec$, respectively. The simulated channel parameters (estimated from real measurements [12], [23]) are $K_{PL} = -41.34$, $n_{PL} = 3.86$, $\xi_{dB} = 3.20$, $\eta = 3.09m$ and $\rho_{dB} = 1.64$. The receiver thermal noise is -110 dBm and the BER threshold is $p_{b,th} = 2\times10^{-6}$. The balancing factor between motion and communication is $\gamma = 0.01$, and the constants C_1, C_2 and C_3 are set to 10, 50 and 10, respectively. The Armijo step size parameters are set to $\alpha = 0.1$ and $\beta = 0.5$. The initial controls $u_0(t)$ and $R_0(t)$ are both set to zero. The integration step size for the simulation is set to $dt = 0.1$ seconds, and the algorithm is run for 500 iterations. The algorithm is terminated whenever k in (11) is greater than 50, indicating that a local minimum has been reached.

The plot for cost (J) versus iteration count is depicted in Fig. 1, showing a rapid decrease in cost during the first few iterations of the algorithm. The cost decreases from the initial value of 2.3872×10^5 to 799.63 in 20 iterations, while the cost after 56 iterations is 565.13 when k became greater than 50, indicating convergence. Fig. 1 also shows the tail of the

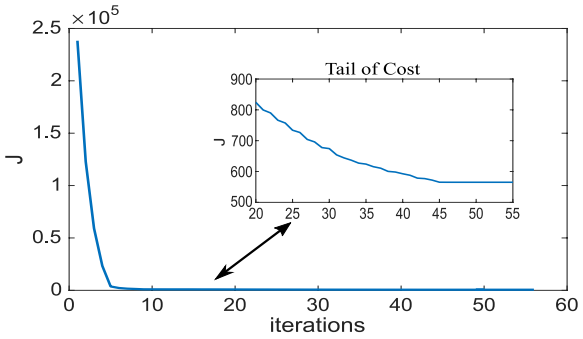

Fig. 1. Cost as function of iteration count.

cost trajectory and evidently it starts flattening after iteration 20. The 56 steps of the algorithm took 0.83 seconds of CPU time on an Intel dual-core computer with i5 processor running at 2.7 GHz.

The total motion and communication cost (6), excluding the penalty term, is $\bar{J} = 475.10$, and the final values of state variables are $x_1(t_f) = (9.8,5)$, $x_2(t_f) = (0.2,-0.8)$, and $x_3(t_f) = 149.7$. We note a mild discrepancy from the desired final values of $x_1(t_f) = (10,5)$, $x_2(t_f) = (0,0)$, and $x_3(t_f) = 150$, It can be reduced by choosing larger penalty terms C_1, C_2, and C_3, however choosing very large penalty terms can degrade the convergence rate. For example, setting $C_1 = 500$, $C_2 = 500$, and $C_3 = 500$ gives $x_1(t_f) = (9.99,5)$, $x_2(t_f) = (-0.08,-0.68)$, and $x_3(t_f) = 149.99$ while the CPU time of the run increased to 7.14 seconds.

Fig. 2 shows the log plot of predicted channel quality metric $(s(x_1) = E[1/\Upsilon(x_1)]$, where $\Upsilon(x_1)$ is the predicted received CNR at position $x_1 = (x,y) \in \mathscr{R}^2)$ and the path taken by the robot in the plane. Smaller values of $s(x_1)$ correspond to good channel quality and vice versa. The robot starting and end positions are marked by a diamond and a square, respectively, in all the figures. Instead of following a straight line between them, the robot takes a detour towards areas with relatively good predicted channel quality. For instance, the point of best channel quality is $q_b = (5,5)$, namely the location of base station, and hence the robot veers towards this point before turning away towards its destination point.

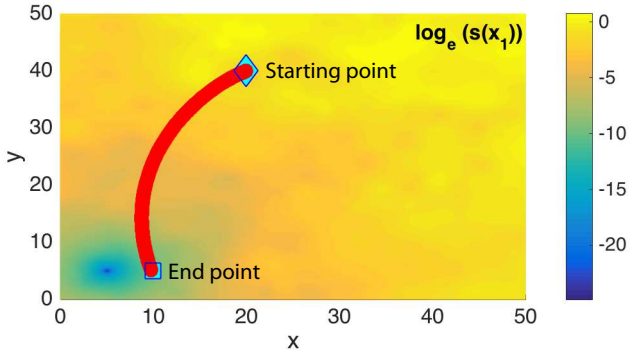

Fig. 2. Path followed by the robot, veering towards regions of better channel quality. Smaller values in the colormap indicate better channel quality prediction as measured by $s(x_1)$.

Fig. 3 depicts a three-dimensional graph of the robot's motion, where the z axis representing time and the motion is in the $x-y$ plane. The upper, blue curve represents the flow of time from 0 to 40 seconds, and the position of the

robot at time t is seen by projecting the corresponding point on the upper curve onto the $x-y$ plane, where it is indicated by a corresponding point on the red curve. Fig. 4 shows the acceleration of the robot along its path, where lengths of the arrows represent its magnitude, and Fig. 5 shows the speed of the robot along its path.

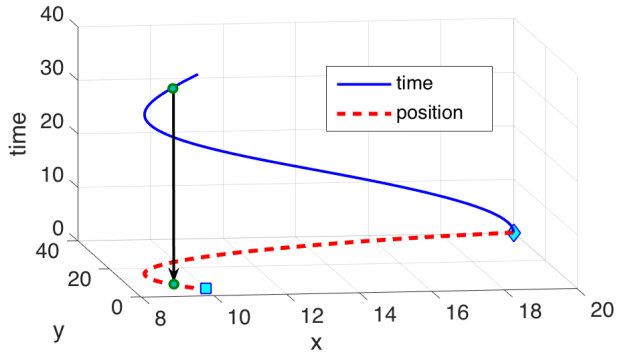

Fig. 3. Position of the robot as a function of time. The diamond and the square indicate the initial and final positions respectively.

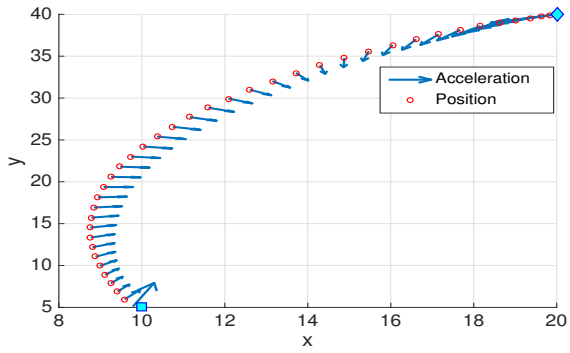

Fig. 4. Acceleration of the robot along its path. The diamond and the square indicate the initial and final positions respectively.

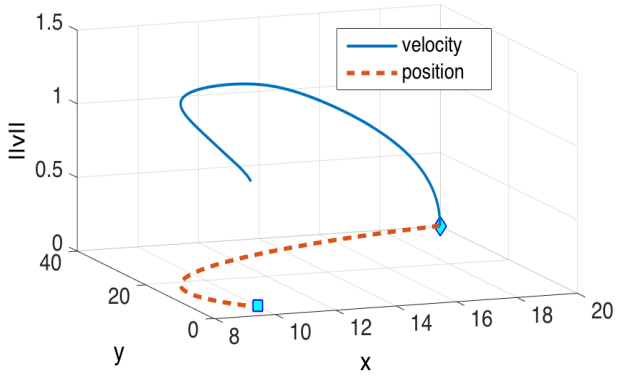

Fig. 5. Velocity of the robot along its path. The diamond and the square indicate the initial and final positions respectively.

The spectral efficiency of the robot along its path is shown in Fig. 6, where the path is marked by red circles, and the spectral efficiency at corresponding points is marked in blue. The 2D map of the predicted channel quality metric $s(x_1)$ is also plotted on the x-y plane (in log scale). It can be seen that the robot transmits with a higher spectral efficiency in regions of better predicted channel quality. This is not surprising since, in regions of higher channel quality, the robot can transmit with a higher rate to the base station with less communication power.

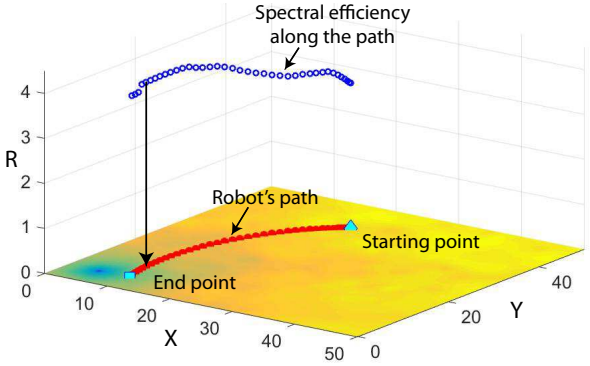

Fig. 6. The robot's spectral efficiency along its path.

IV. Online Optimization

This section extends the algorithm to a realistic and practical online setting, where the robot obtains new channel measurements while in motion. It does not discard the older measurements, but rather appends them by the new data in order to enhance its channel prediction. The online optimal control problem is then to minimize the cost functional

$$J_{t_0} = \int_{t_0}^{t_f} \Big(\frac{2^{R(t)}-1}{K} s(x_1) + \gamma\big(k_1||u(t)||^2 + k_2||x_2(t)||^2 + k_3||x_2(t)|| + k_4 + k_5||u(t)|| + k_6||u(t)||||x_2(t)||\big)\Big) dt + C_1||x_1(t_f) - D||^2 + C_2||x_2(t_f)||^2 + C_3||x_3(t_f) - \bar{c})||^2, \quad (13)$$

subject to the dynamics

$$\begin{aligned} \dot{x}_1(t) &= x_2(t), & x_1(t_0) &= a_1 \\ \dot{x}_2(t) &= u(t), & x_2(t_0) &= a_2 \\ \dot{x}_3(t) &= R(t), & x_3(t_0) &= 0 \end{aligned}$$

and the input constraints (5). Here $t_0 \in [0, t_f]$ is the time at which the optimization is performed, and the terms a_1 and a_2 are the current position and velocity of the robot at time t_0, and $\bar{c} := (c - x_3(t_0^-))$ is the number of bits per unit frequency that remains to be transmitted in the time-interval $[t_0, t_f]$. The online approach solves this problem each time a channel estimation is performed, typically at a finite number of times during the horizon $[0, t_f]$. The initial control point of each such a run of the algorithm consists of the remaining input control computed by its previous run.

The considered problem is the same as the one discussed in Section III, except that the robot performs channel prediction every 10 seconds, and each prediction is based on 100 new channel measurements taken at random locations. Also the initial run, at $t_0 = 0$, solves the offline problem with 100 channel samples. The combined time for channel prediction and a run of the algorithm was about 2 seconds and took under 50 iterations of the algorithm's run.

The results of the simulation are shown in Fig. 7 & 8, where the position of the robot at the end of each predication and optimization cycle (10 seconds) is indicated by a circle. Fig. 7 shows the computed optimal trajectories for each prediction-optimization cycle from the current time to the final time. A concatenation of the computed trajectories, which the robot actually would traverse, is indicated by the red path in Fig. 8, while the dashed blue path indicates the trajectory computed by the offline algorithm at time $t_0 = 0$,

based on the initial channel prediction. The total energy consumed (Eq. (6)) in the offline solution (dashed blue path Fig. 8) is $\bar{J} = 371$, while the solution of the online problem (red path in Fig. 8) yields a lower value, $\bar{J} = 304$.

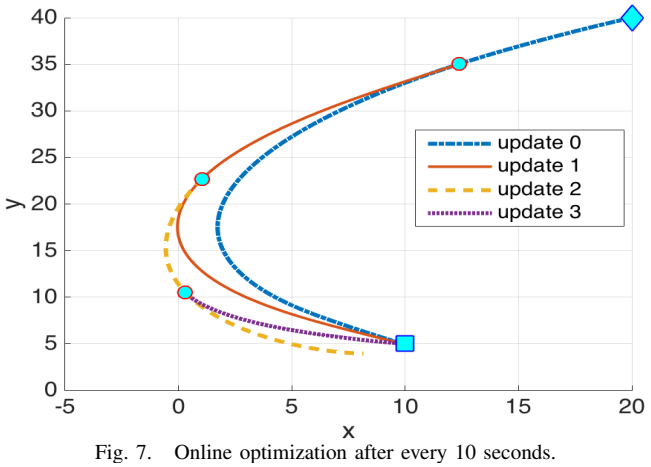

Fig. 7. Online optimization after every 10 seconds.

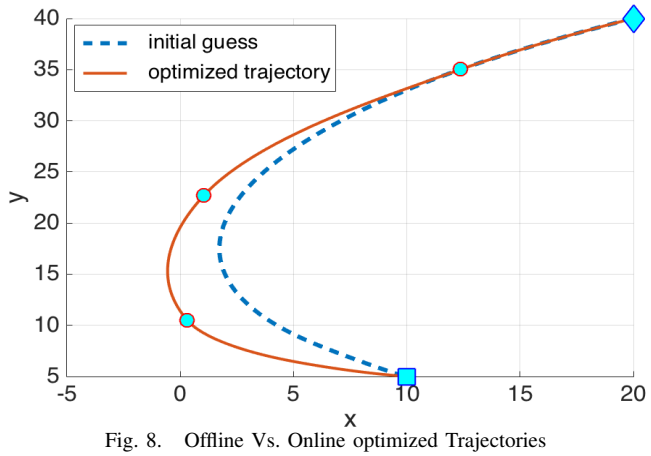

Fig. 8. Offline Vs. Online optimized Trajectories

V. Conclusions and Future Research

We considered the problem of co-optimization of communication and motion power of a robot that is required to transmit a given number of bits to a remote station in a pre-specified amount of time, while moving between a given starting point and an end point. The problem is to compute the robot's optimal path, and acceleration and transmission rate along this path. Both offline and online versions of the problem are considered and solved by simulation in realistic channel environments. Future work would focus on extending the results from the case of a single robot to that of multiple agents having to perform coordinated tasks while maintaining formation in the face of limited energy sources.

References

[1] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, 2002, pp. 1327–1332.

[2] Y. Mostofi, T. Chung, R. Murray, and J. Burdick, "Communication and Sensing Trade Offs in Decentralized Mobile Sensor Networks: A Cross-Layer Design Approach," in *Proceedings of the International Conference on Information Processing in Sensor Networks*, Los Angeles, CA, April 2005, pp. 118–125.

[3] N. E. Leonard, D. Paley, F. Lekien, R. Sepulchre, D. M. Fratantoni, and R. E. Davis, "Collective motion, sensor networks, and ocean sampling," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 48–74, 2007.

[4] F. Bullo, E. Frazzoli, M. Pavone, K. Savla, and S. L. Smith, "Dynamic vehicle routing for robotic systems," *Proceedings of the IEEE*, vol. 99, no. 9, pp. 1482–1504, September 2011.

[5] M. M. Zavlanos, M. B. Egerstedt, Y. C. Hu, and G. J. Pappas, "Graph-theoretic connectivity control of mobile robot networks," *Proceedings of the IEEE*, vol. 99, no. 9, pp. 1525 – 1540, July 2011.

[6] H.-L. Choi, L. Brunet, and J. How, "Consensus-based decentralized auctions for robust task allocation," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 912–926, 2009.

[7] S. L. Smith, M. Schwager, and D. Rus, "Persistent robotic tasks: Monitoring and sweeping in changing environments," *IEEE Transaction on Robotics*, vol. 28, no. 2, pp. 410–426, April 2012.

[8] J. Marden, G. Arslan, and J. Shamma, "Cooperative control and potential games," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 39, no. 6, pp. 1393–1407, 2009.

[9] Y. Mostofi, M. Malmirchegini, and A. Ghaffarkhah, "Estimation of communication signal strength in robotic networks," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Anchorage, Alaska, May 2010, pp. 1946–1951.

[10] A. Ghaffarkhah and Y. Mostofi, "Channel learning and communication-aware motion planning in mobile networks," Jun. 2010, pp. 5413 –5420.

[11] Y. Yan and Y. Mostofi, "Robotic router formation in realistic communication environments," *IEEE Transactions on Robotics*, vol. 28, no. 4, pp. 810 – 827, August 2012.

[12] M. Malmirchegini and Y. Mostofi, "On the spatial predictability of communication channels," *IEEE Transactions on Wireless Communications*, vol. 11, no. 3, pp. 964–978, March 2012.

[13] A. Ghaffarkhah and Y. Mostofi, "Dynamic Networked Coverage of Time-Varying Environments in the Presence of Fading Communication Channels," *ACM Transactions on Sensor Networks*, vol. 10, no. 3, April 2014.

[14] Y. Yan and Y. Mostofi, "Co-optimization of communication and motion planning of a robotic operation under resource constraints and in fading environments," *IEEE Transactions on Wireless Communications*, vol. 12, no. 4, pp. 1562–1572, 2013.

[15] ——, "To go or not to go on energy-aware and communication-aware robotic operation," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 3, pp. 218 – 231, July 2014.

[16] A. J. Goldsmith and S.-G. Chua, "Variable-rate variable-power MQAM for fading channels," *IEEE Transaction on Communications*, vol. 45, no. 10, pp. 1218–1230, October 1997.

[17] P. Tokekar, N. Karnad, and V. Isler, "Energy-optimal trajectory planning for car-like robots," *Autonomous Robots*, pp. 1–22, 2013.

[18] Y. Mei, Y.-H. Lu, Y. C. Hu, and C. S. G. Lee, "Deployment of mobile robots with energy and timing constraints," *IEEE Transaction on Robotics*, vol. 22, no. 3, pp. 507–522, June 2006.

[19] C. C. Ooi and C. Schindelhauer, "Minimal energy path planning for wireless robots," *Mobile Networks and Applications*, vol. 14, no. 3, pp. 309–321, January 2009.

[20] F. El-Moukaddem, E. Torng, G. Xing, and G. Xing, "Mobile relay configuration in data-intensive wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 2, pp. 261–273, 2013.

[21] C. Tang and P. McKinley, "Energy optimization under informed mobility," *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 9, pp. 947–962, 2006.

[22] H. Jaleel, Y. Wardi, and M. Egerstedt, "Minimizing mobility and communication energy in robotic networks: An optimal control approach," in *Proceedings of the American Control Conference*, Portland, OR, 2014, pp. 2662 – 2667.

[23] U. Ali, Y. Yan, Y. Mostofi, and Y. Wardi, "An optimal control approach for communication and motion co-optimization in realistic fading environments," in *Proceedings of the American Control Conference*, 2015, pp. 2930–2935.

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

[25] M. Hale, Y. Wardi, H. Jaleel, and M. Egerstedt, "Hamiltonian-based algorithm for optimal control of switched-mode hybrid systems," in *Technical Memorandum*, Georgia Tech, 2014.

[26] A. E. Bryson and Y.-C. Ho, *Applied optimal control: optimization, estimation and control*. CRC Press, 1975.

[27] E. Polak, *Optimization: algorithms and consistent approximations*. Springer-Verlag New York, Inc., 1997.