2012 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 7-12, 2012. Vilamoura, Algarve, Portugal

A Brain-Machine Interface to Navigate Mobile Robots Along Human-Like Paths Amidst Obstacles

Abdullah Akce, James Norton, and Timothy Bretl

Abstract— **This paper presents an interface that allows a human user to specify a desired path for a mobile robot in a planar workspace with noisy binary inputs that are obtained at low bit-rates through an electroencephalograph (EEG). We represent desired paths as geodesics with respect to a cost function that is defined so that each path-homotopy class contains exactly one (local) geodesic. We apply max-margin structured learning to recover a cost function that is consistent with observations of human walking paths. We derive an optimal feedback communication protocol to select a local geodesic—equivalently, a path-homotopy class—using a sequence of noisy bits. We validate our approach with experiments that quantify both how well our learned cost function characterizes human walking data and how well human subjects perform with the resulting interface in navigating a simulated robot with EEG.**

I. INTRODUCTION

Our work is motivated by the design of non-invasive brain-machine interfaces (BMIs) for the control of robotic systems that include humanoid robots [1], [2], wheelchairs [3]–[6], and other mobile robots [7], [8]. These interfaces translate electroencephalograph (EEG) recordings of brain activity into desired commands for a robot, effectively enabling people to control robots just by thinking. In most of these interfaces, possible commands for the robot were either pre-determined high-level tasks (e.g., go to kitchen) that the robot could perform autonomously [1], [4], or were low-level steering tasks (e.g., turn left/right) that defined the moment-to-moment course of the robot [2], [3], [6]. In particular, the level of task specification was an important design choice and it was critical to the overall performance of the interface [9]. In this work, we take a different approach and allow users to specify a human-like path for the robot with a sequence of binary commands.

We have drawn considerable inspiration from the success of [5] in enabling a human user to drive a robotic wheelchair indoors on level ground using EEG signals. Their approach was to drive the wheelchair autonomously along a sequence of locations chosen by the human user. The interface showed the map of the environment visible from the wheelchair, and the user chose a location from a set of locations indicated on this map. Although their interface was successful, the paths specified by the user did not correspond to paths that humans might prefer for indoor navigation, and the navigation was slow in comparison to conventional input devices, e.g., a joystick.

A. Akce is with the Department of Computer Science, J. Norton is with the Neuroscience Program, and T. Bretl is with the Department of Aerospace Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, 61801, USA {aakce2,jnorton4,tbretl}@illinois.edu

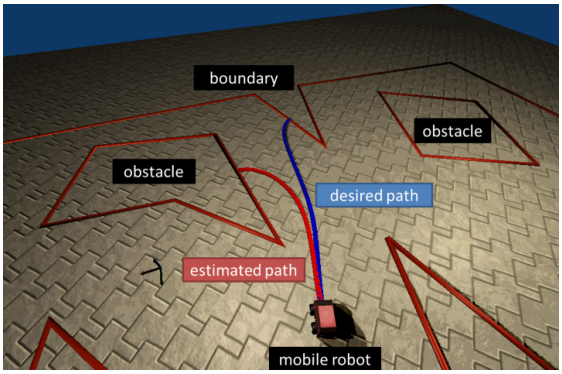


Fig. 1. Our interface for navigating a mobile robot in a planar workspace with polygonal obstacles. The interface allows a human user to specify a desired path, which corresponds to a (local) geodesic from the robot's current location to a boundary point. The interface provides feedback by showing its estimate of the user's desired path (called estimated path). The user provides binary inputs by determining the (clockwise) ordering of their desired path with respect to the estimated path. See text for details.

It is necessary to restrict the space of all possible paths the user can specify because we cannot describe arbitrary paths with a finite number of inputs. In previous work [10], we made a heuristic choice, and used an ordered symbolic language to represent paths of piecewise-constant curvature. However, this decision made it hard to incorporate certain types of statistical information. For example, how does path likelihood vary in the presence of obstacles? Recently [11], we suggested a more systematic approach, observing that the principle of optimality could be used to generate a compact representation of all paths likely to be seen in the context of a particular application. We restricted our scope to only paths that are locally-shortest and showed that the space of all such paths having length that is bounded and locally-minimal in a polygonal workspace with polygonal obstacles is homeomorphic to a unit disk. This homeomorphism was used to design an interface by which the user can, with vanishing error probability, specify a locally-shorted path to a boundary point of the workspace. Although the applicability of this approach to the design of a BMI for navigating a simulated robot amidst obstacles was considered, its scope was limited because the chosen optimality principle (minimizing length) caused paths to touch obstacles, and user inputs were provided by a keyboard rather than by EEG signals.

This paper presents an interface that allows users to navigate a mobile robot along human-like paths in a simulated environment with input only from EEG (Fig. 1). In particular, we represent a desired path as a (local) geodesic with

978-1-4673-1736-8/12/S31.00 ©2012 IEEE

respect to a cost function that is defined so that each path-homotopy class contains exactly one geodesic (Section II-A). We recover such a cost function from a dataset of human-demonstrated paths using max-margin structured learning (Section II-B). We represent path-homotopy classes in a compact way by noting that the space of all path-homotopy classes originating from a fixed starting point is homeomorphic to the unit disk (Section II-C). This representation is used to design a feedback communication protocol that allows a user to select a desired geodesic, with vanishing error probability, using a sequence of noisy binary inputs (Section II-D). Section III describes the implementation of our interface for navigating a simulated robot with EEG. We then describe the experiments used to learn the cost function from a dataset of human-walking paths (Section IV-A), and the experiments performed to evaluate the performance of the resulting interface in navigating a simulated robot using EEG (Section IV-B). Section V concludes the paper.

II. METHODS

A. Representing Desired Paths as Geodesics

We represent desired paths as geodesics that are locally-optimal solutions to

$$\begin{aligned} \text{minimize} \quad & \int_{t=0}^{t=T} g(\gamma(t))dt \\ \text{s.t.} \quad & \gamma(t) \in Q_{\text{free}}, \forall t \in [0,T] \\ & \gamma(0) = q_0 \text{ and } \gamma(T) = q_1, \end{aligned} \tag{1}$$

where $\gamma : [0,T] \to Q_{\text{free}}$ is a continuous function, Q_{free} is the free configuration space, $g : Q_{\text{free}} \to \mathbb{R}^+$ is a given cost function, q_0 and q_1 are given start and end configurations, respectively, and T is the free final time. To make things concrete, we consider a point robot moving through a bounded planar workspace with polygonal obstacles, which has $Q_{\text{free}} \in \mathbb{R}^2$. Obstacles induce a topological structure to paths in Q_{free}. We say that two paths γ and γ' are homotopic if we can continuously deform one to another. The homotopy defines an equivalence relation and this relation divides paths into path-homotopy classes. We assume that a cost function $g : Q_{\text{free}} \to \mathbb{R}^+$ is given so that there is a unique geodesic for each path-homotopy class. This allows us to represent a geodesic only by a path-homotopy class which may be defined using a (reference) path $\pi : [0,1] \to Q_{\text{free}}$ with $\pi(0) = q_0$ and $\pi(1) = q_1$, and denoted $[\pi]$.

B. Learning an Optimality Principle from Data

In order to generate human-like paths, we learn a cost function so that the resulting geodesics will resemble paths that humans prefer in navigating a mobile robot (e.g., a wheelchair) amidst obstacles in a target path-homotopy class. Our approach is to use maximum-margin structured learning (MMSL) [12], [13] to recover such a cost function from a training set $D = \{\gamma^i\}_{i=1}^N$ where $\gamma^i : [0,T] \to Q_{\text{free}}$ is a human-demonstrated path. We assume that the cost function $g : Q_{\text{free}} \to \mathbb{R}^+$ takes the form of $g(q) = w^T c(q)$ where w is a non-negative weight vector in a convex parameter space W, and $c = [c_1, \ldots, c_n]$ is a given vector-valued feature function, with each $c_i : Q_{\text{free}} \to \mathbb{R}^+$. The total cost of a path $\gamma : [0,T] \to \mathbb{R}^2$ in Γ can be expressed as $J(\gamma) = \int w^T c(\gamma(t))dt = w^T f(\gamma)$, where $f : \Gamma \to \mathbb{R}^+$ is the feature sum along the path γ. In MMSL, we introduce a margin that scales with the loss of choosing an alternative $\gamma \in \Gamma$ in place of the demonstrated example γ^i, and denote it by the function $L_i : \Gamma \to \mathbb{R}^+$, where $L_i(\gamma) > 0$ for all $\gamma \neq \gamma^i$, and $L_i(\gamma^i) = 0$. The underlying problem is to learn weights $w \in W$ so that

$$w^T f(\gamma^i) \leq w^T f(\gamma) - L_i(\gamma), \quad \forall y \neq y^i, \forall i,$$

which requires the cost of an alternative γ to be larger than the cost of the example γ^i. Maximizing margin subject to these constraints is equivalent to the convex program

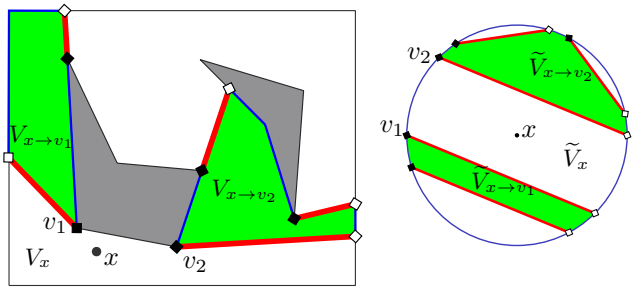
$$\begin{aligned} \min_{w \in W, \zeta_i \in \mathbb{R}} \; & \frac{1}{N}\sum_{i=1}^N \zeta_i + \frac{\lambda}{2}\|w\|^2 \\ \text{s.t. } & w^T f(\gamma^i) \leq \min_{\gamma \in \Gamma} \left\{ w^T f(\gamma) - L_i(\gamma) \right\} + \zeta_i, \quad \forall i, \end{aligned}$$

where $\{\zeta_i\}_{i=1}^N$ are the slack variables, and $\lambda \geq 0$ is a constant that trades off between a penalty on constraint violations and margin maximization. This program can be solved using a subgradient descent algorithm [12], [13].

C. A Compact Representation for Path-Homotopy Classes

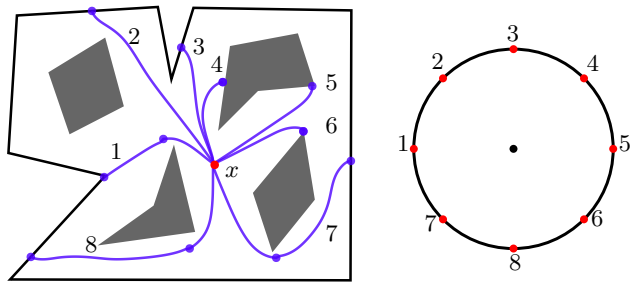
In order represent a path-homotopy class in a compact way, we construct a homeomorphism between the space of all path-homotopy classes and the unit disk. Our approach is to use the homeomorphism, previously derived in [11], between the space of all locally-shortest paths originating from a fixed point and the unit disk. This homeomorphism is constructed recursively by iterating through regions of the workspace that become visible after each iterative movement of the robot along the shortest path segment. Such a region $V \in Q_{\text{free}}$ has a boundary consisting of polygonal chains in the interior of Q_{free}, called *extension edges*, and polygonal chains in the boundary of Q_{free}, called *boundary edges*. The algorithm embeds V into a cell of the unit disk, denoted $\tilde{V}$, such that the boundary of $\tilde{V}$ contains a chord for each extension edge of V, and a circular arc on the disk boundary for each boundary edge of V. This process is illustrated in Fig 2, and described in [11].

The space of all path-homotopy classes in $Q_{\text{free}} \in \mathbb{R}^2$ with a starting point at q is known as the universal covering space $\bar{Q}$ denoted by $\bar{Q} = \{[\gamma] | \gamma : [0,1] \to Q_{\text{free}} \text{ and } \gamma(0) = q\}$. We can think of $\bar{Q}$ as the space formed by concatenating the end points of all path-homotopy classes from q, hence each point in $\bar{Q}$ represents a unique path-homotopy class. Let $\bar{Q}_b \subset \bar{Q}$ be the set of path-homotopy classes that terminate at a boundary point of the workspace. Since there is a unique locally-shortest path in each path-homotopy class, the homeomorphism derived in [11] gives us a one-to-one mapping between $\bar{Q}$ and the closed unit disk $D^1 = \{x | x \in \mathbb{R}^2, \|x\| \leq 1\}$ such that the restriction of this mapping to $\bar{Q}_b$ is the boundary of the unit disk, denoted by S^1.



(a) Sample visibility regions (b) Their embeddings in D^1

Fig. 2. One iteration of constructing a homeomorphism between locally-shortest paths originating at x and the closed unit disk D^1. V_x is the visibility polygon of x, and $V_{p\to q}$ is the set of points that become newly visible by moving from p to q. Their embeddings are $\widetilde{V}_x$ and $\widetilde{V}_{p\to q}$, respectively. Extension edges are in red, boundary edges are in blue.



(a) Sample geodesics (b) Their representation in S^1

Fig. 3. A set of geodesics starting at x and ending at the boundary are shown in (a), their corresponding points in the unit circle S^1 are shown in (b). The intermediate points along the geodesics in (a) indicate the points at which the geodesics cross an extension edge.

There are three properties of this mapping that will allow us to design an optimal feedback communication protocol for selecting desired geodesics in Section II-D. Let Γ be the space of all geodesics from a fixed starting point, , Γ_b be the set of geodesics in Γ that terminate at a boundary point, and $\phi : \Gamma \to D^1$ be the homeomorphism between Γ and D^1. The first property is that any geodesic $\gamma_b \in \Gamma_b$ that crosses the sequence of extension edges $e_1, \ldots, e_i$ is mapped to an angle $\theta = \phi(\gamma_b)$ such that $\phi(\gamma_l) < \theta < \phi(\gamma_r)$, where γ_l, γ_r are the geodesics that cross the sequence of extension edges $e_1, \ldots, e_{i-1}$ and terminate at an endpoint of e_i at the boundary. This allows us to efficiently query the probability that a geodesic crosses a particular extension edge. The second property is that any geodesic in Γ is a prefix of some other geodesic in Γ_b. With this property, navigation along any path $\gamma \in \Gamma$ can be accomplished by carrying out the navigation along the path $\gamma_b \in \Gamma_b$ with prefix γ, and then by stopping the navigation when the desired endpoint $\gamma(1)$ is reached. The third property is that we can induce an ordering between the geodesics in Γ_b using their representation in S^1. We say that $\gamma_1 \in \Gamma_1$ is ordered to the left of γ_2, denoted $\gamma_1 < \gamma_2$, if and only if the clockwise angle from $\phi(\gamma_1)$ to $\phi(\gamma_2)$ is smaller than the counterclockwise angle from $\phi(\gamma_1)$ to $\phi(\gamma_2)$, as illustrated in Fig. 3. A human user may easily determine the ordering of two geodesics in Γ_b because these geodesics cannot cross each other in $\bar{Q}$.

D. A Communication Protocol for Selecting Geodesics

In this section, we describe an optimal feedback communication protocol that allows a human user to select a geodesic $\gamma^* \in \Gamma_b$, or equivalently the angle $\theta^* = \phi(\gamma^*)$, with vanishing error probability, using a sequence of noisy binary inputs. Such a protocol says how the user must choose inputs and how the interface must provide feedback so that $E(|\phi(\widehat{\gamma}_k) - \phi(\gamma^*)|) \to 0$ as $k \to \infty$, where $\widehat{\gamma}_k$ is the estimate of γ^* after k user inputs. Here, we use the protocol derived in [11] to communicate an angle across a binary symmetric channel (BSC) with noiseless feedback. We model the noisy input source as a BSC with crossover probability ϵ. At time step k, the input to this channel is $x_k \in \{0, 1\}$, where we associate $x_k = 0$ with the input "left" and $x_k = 1$ with the input "right". The output of the channel is $y_k \in \{0, 1\}$ with $P(Y_k \neq X_k) = \epsilon$, where Y_k, X_k are random variables corresponding to channel input and output, respectively. We assume that the BSC can provide noiseless feedback to the user, which in our case corresponds to providing an estimate $\widehat{\gamma}$ of the user's desired path γ^* to the user and assuming that the user can determine with perfect accuracy whether or not $\gamma^* < \widehat{\gamma}$ according to the ordering defined in Section II-C.

Our protocol is as follows. Assume that at time step k, the interface computed the posterior distribution $P_{\Theta|Y^k}(\theta|y_1 \ldots y_k)$, where $\Theta \in S^1$ is the random variable indicating the desired angle, and $Y_k = (Y_1, \ldots, Y_k)$. First, the interface finds a pair of angles $(\mu_k, \bar{\mu}_k)$ that are opposite to each other in S^1, i.e., $\bar{\mu}_k = (\mu_k + \pi) \mod (2\pi)$, and that the probability concentrated on the half circle from μ_k to $\bar{\mu}_k$ is equal to the probability concentrated on the opposite half circle from $\bar{\mu}_k$ to μ_k. The interface selects the angle in $(\mu_k, \bar{\mu}_k)$ with higher posterior density as the estimate $\widehat{\theta}_k$, and provides it as feedback by showing the geodesic $\widehat{\gamma}_k = \phi^{-1}(\widehat{\theta}_k)$. Then, the user selects the next input x_{k+1} as 1 if $\gamma^* \geq \widehat{\gamma}_k$ or as 0 otherwise. Finally, if $y_{k+1} = 1$ (the case $y_{k+1} = 0$ is analogous), then the interface applies Bayes' rule to update the posterior distribution as

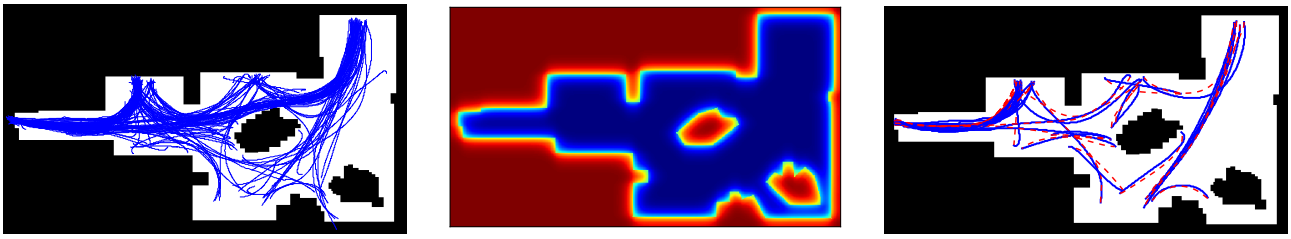
$$P_{\Theta|Y^{k+1}}\left(\theta \middle| y_1 \ldots y_{k+1}\right) = \eta \cdot \begin{cases} (1-\epsilon) \cdot P_{\Theta|Y^k}\left(\theta \middle| y_1 \ldots y_k\right) & \text{if } \phi^{-1}(\theta) \geq \widehat{\gamma}_k \\ \epsilon \cdot P_{\Theta|Y^k}\left(\theta \middle| y_1 \ldots y_k\right) & \text{otherwise,} \end{cases} \quad (2)$$

where η is a normalizing constant. Then, the process repeats.

III. Interface for Navigating a Simulated Robot with EEG

A. Our Approach for Navigation with Binary Inputs

In this section, we describe our algorithm for enabling the navigation of the robot while the user is specifying their desired path by following the communication protocol derived in Section II-D. Let $q_0 \in Q_{\text{free}}$ be the robot's starting position, and $\bar{Q}$ be the universal covering space consisting of all path-homotopy classes with fixed start point q_0. At time step k, the interface obtains the noisy input y_k from the user and computes the posterior distribution



(a) Intel pedestrian path prediction dataset (b) The recovered cost function (c) Test paths and corresponding geodesics

Fig. 4. The dataset consisting of human walking paths is shown on a 525 by 325 map in (a), where black pixels denote obstacles, and white pixels denote free space. The recovered cost function is shown in (b), where red pixels denote a high cost and blue pixels denote a low cost. The paths in the test dataset (dashed blue curves) and the geodesics (solid blue curves) generated with the recovered cost function for each test path is shown in (c).

$P_{\Theta|Y^k}(\theta|y_1,\ldots,y_k)$. From this distribution, it generates the estimate $\widehat{\theta}_k \in S^1$ and the geodesic $\widehat{\gamma}_k = \phi^{-1}(\widehat{\theta}_k)$ that starts at q_0 and ends at a boundary point. The estimate $\widehat{\theta}_k$ also describes a path-homotopy class with an endpoint q_f in $\bar{Q}$. Instead of displaying $\widehat{\gamma}_k$ as feedback, the interface displays the geodesic $\widehat{\pi}_k$ from the robot's current position $q \in \bar{Q}$ to $q_f \in \bar{Q}$. Let e be the first extension or boundary edge that $\widehat{\pi}$ crosses in $\bar{Q}$ (see Section II-C). The robot moves along $\widehat{\pi}$ until it crosses e when the posterior probability of the event that the user's desired path crosses e is at least 0.90. This procedure is outlined in Algorithm 1.

Algorithm 1 Robot Navigation Algorithm.

let $\bar{Q}$ be the universal covering space
while robot is not at a boundary point **do**
  wait until noisy input y_k is observed
  evaluate $P_{\Theta|Y^k}$ and obtain $\widehat{\theta}_k \in S^1, \widehat{\gamma}_k \in \Gamma_b$
  let q be the robot's current position in $\bar{Q}$
  let q_f be the point in $\bar{Q}$ corresponding to $\widehat{\theta}_k$
  display geodesic $\widehat{\pi}_k$ from q to q_f in $\bar{Q}$ as feedback
  let e be the first edge that $\widehat{\pi}_k$ crosses in $\bar{Q}$
  if posterior probability of crossing $e \geq 0.90$ **then**
    command robot to move along π_k until it crosses e
  else
    command robot to stop
  end if
end while

B. Our Approach for Obtaining Binary Inputs through EEG

Our BMI was based on steady-state visually-evoked potentials (SSVEP), a natural neural response to repetitive flickering stimuli in the environment [14]. By providing stimuli of known frequency patterns, it is possible to determine which stimulus the user is attending to. Our interface displayed two stimuli that steadily flashed on a CRT monitor at 8.67Hz and 12Hz. These frequencies were chosen because they yield high signal-to-noise ratio and lie outside of the range known to induce seizures [15]. EEG signals were extracted from seven electrode sites across the occipital region of the scalp, in particular PO7, PO3, PO4, PO8, O1, OZ, O2, at impedances not exceeding 10kΩ, with a reference measured at PZA [16]. These signals were acquired using a 128-channel bioamplifier at 256Hz, bandpass-filtered from 1Hz to 30Hz, and analyzed with BCI2000 [17] and MATLAB using a three second sliding window. The signals from each channel were filtered into four different spatial representations using bipolar and Laplacian configurations [18]. For each spatial filter, we computed a signal to noise ratio (SNR) of each frequency of interest together with its first harmonic. For each frequency, after discarding the highest and lowest SNR values, the average of the remaining two SNR values was obtained. A classification was made when this average exceeded a threshold of 8. After each classification, auditory feedback was provided to the user by playing a unique sound to indicate observation of a "left" or "right" input.

IV. EXPERIMENTS

A. Learning Cost Function from Data

We performed experiments to learn the cost function in (1) using the Intel pedestrian path prediction dataset [19] that consists of human walking paths recorded in an office environment (Fig. 4a). Similar to the procedure in [19], a black and white pixel map of the environment was generated, with a pixel corresponding to a distance of 0.04 meters. Ten cost features were defined for each pixel based on blurred images of the pixel map with different levels of blurring. The loss function, a measure of dissimilarity of a given path γ from an example path γ_i, was computed by first assigning a zero loss to all pixels that are within 7 pixels (0.28 meters) of distance from γ_i, and constant loss to all other pixels, and then by summing the loss values assigned to the pixels that γ crossed over.

We observed that some paths in the dataset do not resemble any optimal behavior that can be explained by our cost features. We labeled such paths as outliers and removed them from the dataset. In order to detect whether a path γ_i was an outlier, we applied MMSL using γ_i as the only training example to obtain a cost function g_i, and computed the loss of the geodesic from the start point $\gamma_i(0)$ to the end point $\gamma_i(1)$ under the cost function g_i. If this loss was non-zero, we labeled the path γ_i as an outlier. Out of 166 paths, 100 paths were found to be outliers, and the remaining paths were split into a training set and a test set, each containing 33 paths.

The recovered cost function with MMSL using all paths in the training set is illustrated in Fig. 4b. We empirically verified that geodesics generated with respect to this cost

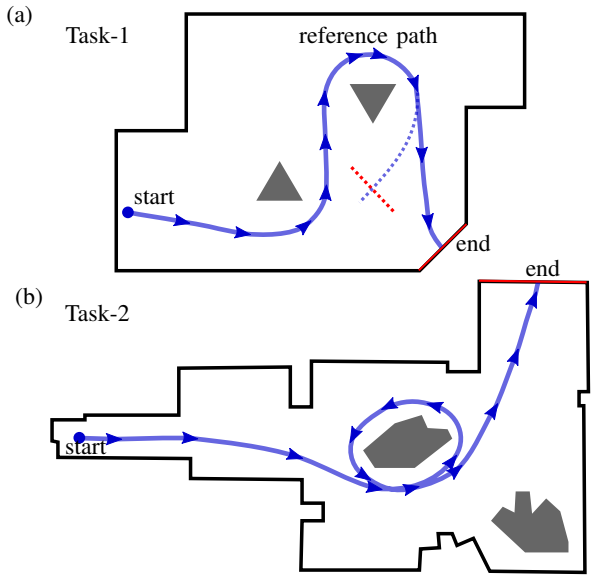


Fig. 5. Description of the two experimental tasks. The goal was to navigate the robot from start (blue dot) to the midpoint of the finish segment (red boundary segment) by specifying a desired path homotopic to the reference path (blue curve). Task-1 was very similar to the task used in [5], where the reference path followed the dashed blue curve towards the end.

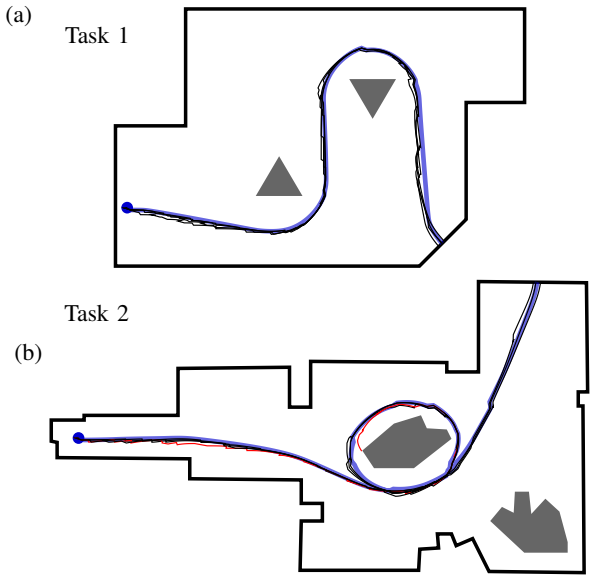


Fig. 6. The actual paths (thin curves) followed by the mobile robot compared to the geodesic homotopic to the reference path (thick blue curve). The black curves correspond to the resulting paths from the successful trials. The red curve corresponds to the resulting path obtained in the failed trial.

TABLE I
EXPERIMENT RESULTS FOR TASK 1

	Subject A		Subject B	
	Trial 1	Trial 2	Trial 1	Trial 2
Task success	YES	YES	YES	YES
Time-to-navigate (s)	177	188	142	138
Time-opt-ratio	1.69	1.79	1.35	1.31
Time-to-specify (s)	143	102	117	99
Input accuracy	0.83	0.82	1.0	0.95
Input latency (s)	3.4	3.7	6.2	5.0
ITR (bits/min)	6.04	5.19	9.68	8.56

TABLE II
EXPERIMENT RESULTS FOR TASK 2

	Subject A		Subject B	
	Trial 1	Trial 2	Trial 1	Trial 2
Task success	YES	NO	YES	YES
Time-to-navigate	253	-	257	314
Time-opt-ratio	1.62	-	1.65	2.01
Time-to-specify	138	-	165	264
Input accuracy	0.91	0.82	1.0	0.88
Input latency	4.3	8.1	6.9	8.0
ITR (bits/min)	7.86	2.37	8.70	3.53

function in each path-homotopy class were unique. For evaluation, we computed the loss of the geodesics corresponding to each training and test path. In the training set, 45% of the geodesics had zero-loss and in the test set, 55% of the geodesics had zero-loss. The results (Fig. 4c) suggest that we can approximate human walking paths with zero-loss by a geodesic curve in about half of the cases, and the recovered cost function generalizes well to new cases.

B. Using the Interface to Navigate a Simulated Robot

In order to quantify the performance of our interface for navigating a simulated robot with EEG, we performed experiments with two able-bodied subjects that has no prior experience in SSVEP-based BMIs. Each subject completed two trials of two unique navigation tasks (Fig. 5). The goal in each task was to navigate the robot from its start position to a goal position in the boundary by specifying a desired path homotopic to a target reference path. In task-1, we simulated the environment used in [5] to evaluate their EEG-based BMI for navigating a physical wheelchair. Our reference path was very similar to theirs except that our path terminated in the boundary rather than in the free space. In task-2, we simulated the environment in which the dataset of human walking paths used in Section IV-A was produced.

We used the following metrics to evaluate the performance of our brain-machine interface:

- **Task success**: whether the robot successfully navigated along a path that was homotopic to the reference path.
- **Time-to-navigate**: the duration of the task in seconds, if the task was successful.
- **Time-opt-ratio**: the ratio of time-to-navigate to the time it would take for the robot to continuously move along the geodesic homotopic to the reference path, which was 105 seconds for task-1, and 156 seconds for task-2.
- **Time-to-specify**: the time it took for the posterior probability of choosing a geodesic with an end point in the finish segment to exceed 90% probability.
- **Input accuracy**: the fraction of inputs that were correctly classified before time-to-specify.
- **Input latency**: the average time in seconds between two consecutive inputs.
- **ITR**: the information transfer rate that gives the number

of reliable bits obtained from the user per minute.

Results show that our interface enabled subjects to successfully complete the navigation tasks in 7 of the 8 trials. The actual paths followed by the mobile robot are shown in Fig. 6, and the performances obtained in task-1 and task-2 are reported in Table I, and II, respectively. Remarkably, in all successful trials, the actual paths were very close to the geodesics homotopic to the reference paths. Subjects obtained between 5 and 10 bits/minute ITR in all trials except in trial-2 of task-2, where both subjects reported being tired and unable to focus their attention during the entire time. This might explain why subject A failed in trial-2 of task-2.

In task-1, the average time-to-navigate was 2.7 minutes, which is significantly less than the average time-to-navigate of 9.5 minutes reported in [5]. However, the fact that they used a different paradigm (P300) for obtaining inputs from EEG than ours (SSVEP), the fact that they used a physical wheelchair rather than a simulated mobile robot, and the fact that they did not rely on a given map of the environment prevents us from making a true comparison. For both tasks, time-opt-ratio was between 1.3 and 2.0 (time-opt-ratio of [5] was 5.4), which means that the navigation time with our interface was less than twice the time it would have taken the robot to move continuously along the geodesic homotopic to the reference path. It is important to note that the average gap between time-to-specify and time-to-navigate was 1.0 minutes, meaning that the subject specified a geodesic with an end point in the finish segment of the task about one minute before the robot actually moved to the midpoint of the finish segment. This gap was partly because between two consecutive user inputs, the robot was only allowed to move until it crossed an extension edge (see Algorithm 1). In future work, the robot might be allowed to move along a longer prefix of the specified geodesic and the speed of the robot might be adjusted to reduce this gap.

The video submission demonstrates our experimental setup and recorded screencasts from subject A's trial-2 of task-1 and trial-1 of task-2.

V. CONCLUSION

This study demonstrated an interface that allows users to navigate a robot through a planar space containing obstacles using only the inputs from an SSVEP-based BMI. By representing desired paths as geodesics under a cost function recovered from human-demonstrated paths, we enabled users to navigate this space in a smooth human-like manner with only binary inputs. Our results suggest that not only can users navigate in this manner, but that they can do so with a very high success rate. Our subjects navigated the robot along two experimental paths in less than twice the time it would have taken the robot if informed of the path explicitly before the task. In the future, we would like to use a 3-class SSVEP-based BMI to enable the user to start or stop the robot at their will. Optimization of the robot's control behaviors and our signal classification algorithm may allow us to further improve the performance.

VI. ACKNOWLEDGMENTS

This work was supported by the National Science Foundation under CPS-0931871 and Grant CMMI-0956362.

REFERENCES

[1] C. J. Bell, P. Shenoy, R. Chalodhorn, and R. P. N. Rao, "Control of a humanoid robot by a noninvasive brain-computer interface in humans," *Journal of Neural Eng.*, vol. 5, no. 2, pp. 214–220, 2008.

[2] M. Chung, W. Cheung, R. Scherer, and R. Rao, "A hierarchical architecture for adaptive brain-computer interfacing," in *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

[3] J. Millan, F. Renkens, J. Mouriño, and W. Gerstner, "Noninvasive brain-actuated control of a mobile robot by human EEG," *Biomedical Eng., IEEE Tran. on*, vol. 51, no. 6, pp. 1026–1033, 2004.

[4] B. Rebsamen, C. Guan, H. Zhang, C. Wang, C. Teo, M. Ang, and E. Burdet, "A brain controlled wheelchair to navigate in familiar environments," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 18, no. 6, pp. 590–598, 2010.

[5] I. Iturrate, J. Antelis, A. Kubler, and J. Minguez, "A noninvasive brain-actuated wheelchair based on a P300 neurophysiological protocol and automated navigation," *Robotics, IEEE Transactions on*, vol. 25, no. 3, pp. 614–627, June 2009.

[6] F. Galan, M. Nuttin, E. Lew, P. Ferrez, G. Vanacker, J. Philips, and J. del R. Millan, "A brain-actuated wheelchair: Asynchronous and non-invasive brain-computer interfaces for continuous control of robots," *Clinical Neurophysiology*, vol. 119, no. 9, pp. 2159 – 2169, 2008.

[7] X. Perrin, R. Chavarriaga, F. Colas, R. Siegwart, and J. d. R. Millán, "Brain-coupled interaction for semi-autonomous navigation of an assistive robot," *Robot. Auton. Syst.*, vol. 58, pp. 1246–1255, 2010.

[8] A. Royer, A. Doud, M. Rose, and B. He, "EEG control of a virtual helicopter in 3-dimensional space using intelligent control strategies," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 18, no. 6, pp. 581 –589, 2010.

[9] J. R. Wolpaw, "Brain-computer interfaces as new brain output pathways," *The Journal of Physiology*, vol. 579, no. 3, pp. 613–619, 2007.

[10] A. Akce, M. Johnson, and T. Bretl, "Remote teleoperation of an unmanned aircraft with a brain-machine interface: Theory and preliminary results," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, May 2010, pp. 5322 –5327.

[11] A. Akce and T. Bretl, "A compact representation of locally-shortest paths and its application to a human-robot interface," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 2011.

[12] N. D. Ratliff, A. J. Bagnell, and M. A. Zinkevich, "Maximum margin planning," in *ICML '06: Proceedings of the 23rd international conference on Machine learning*. New York, NY, USA: ACM, 2006, pp. 729–736.

[13] N. Ratliff, J. A. Bagnell, and M. Zinkevich, "Subgradient methods for maximum margin structured learning," in *ICML Workshop on Learning in Structured Output Spaces*, 2006.

[14] G. Muller-Putz, R. Scherer, C. Brauneis, and G. Pfurtscheller, "Steady-state visual evoked potential (SSVEP)-based communication: impact of harmonic frequency components," *Journal of Neural Engineering*, vol. 2, p. 123, 2005.

[15] R. Fisher, G. Harding, G. Erba, G. Barkley, and A. Wilkins, "Photic- and pattern-induced seizures: A review for the epilepsy foundation of america working group," *Epilepsia*, vol. 46, no. 9, pp. 1426–1441, 2005.

[16] V. Jurcak, D. Tsuzuki, and I. Dan, "10/20, 10/10, and 10/5 systems revisited: their validity as relative head-surface-based positioning systems," *Neuroimage*, vol. 34, no. 4, pp. 1600–1611, 2007.

[17] G. Schalk, D. McFarland, T. Hinterberger, N. Birbaumer, and J. Wolpaw, "BCI2000: a general-purpose brain-computer interface (bci) system," *Biomedical Engineering, IEEE Transactions on*, vol. 51, no. 6, pp. 1034–1043, 2004.

[18] O. Friman, I. Volosyak, and A. Graser, "Multiple channel detection of steady-state visual evoked potentials for brain-computer interfaces," *Biomedical Engineering, IEEE Transactions on*, vol. 54, no. 4, pp. 742–750, 2007.

[19] B. D. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. A. Bagnell, M. Hebert, A. K. Dey, and S. Srinivasa, "Planning-based prediction for pedestrians," in *Intelligent Robots and Systems, 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 3931–3936.