Robust Communication with IoT Devices using Wearable Brain Machine Interfaces

Md Muztoba, Ujjwal Gupta, Tanvir Mustofa and Umit Y. Ogras
School of Electrical, Computer, and Energy Engineering, Arizona State University

Abstract—**Proliferation of internet-of-things (IoT) will lead to scenarios where humans will interact with and control a variety of networked devices including sensors and actuators. Wearable brain-machine interfaces (BMI) can be a key enabler of this interaction for people with disabilities and limited motor skills. At the same time, BMI can improve the experience of healthy individuals significantly. However, state-of-the-art BMI systems have limited applicability as they are prone to errors even with sophisticated machine learning algorithms used for classifying the electroencephalogram (EEG) signals. We improve the reliability of BMI communication significantly by proposing two techniques at higher abstraction layers. Our first contribution is a command confirmation protocol that protects the brain-machine communication against false interpretations at run time. The second contribution is an off-line optimal event selection algorithm that identifies the most reliable subset of events supported by the target BMI system. The event selection is guided by novel user specific reliability metrics defined for the first time in this paper. Extensive experiments using a commercial BMI system demonstrate that the proposed techniques increase the communication robustness significantly, and reduce the time to complete a complex navigation task by 63% on average.**

I. Introduction

IoT vision enables interconnection of a massive number of addressable smart objects or "things" capable of sensing, actuation, and communication [1, 26]. These smart things will not only be interconnected among each other, but they will also interact with humans [6]. Since the source of information is the brain, it is natural to consider brain-machine interfaces as a promising mechanism to convey the human intent to IoT devices. Indeed, wearable BMI systems can provide healthy individuals with a convenient communication pathway to multiple objects in the environment, such as home appliances, sensors, smart cars, and robots. More importantly, BMI may be the only communication choice for people who have lost their motor skills due to accidents, disabilities like amyotrophic lateral sclerosis (ALS), or aging.

Current BMI systems can be classified as invasive and non-invasive. In invasive systems, electrodes are planted inside the brain or right under the skull, while the non-invasive systems rely on external electrodes that touch the scalp [13]. Human intentions are first captured by the EEG signals sensed by the electrodes. Then, the EEG signals are compared to the training data of known events using sophisticated classification algorithms to identify emotions, facial expressions, and cognitive inputs. Wearable BMI, such as the Emotiv neuro-headset [7] used in this work, is a subset of the non-invasive systems where the user does not have to stay connected to an experimental platform through wired electrodes, as depicted in Figure 1.

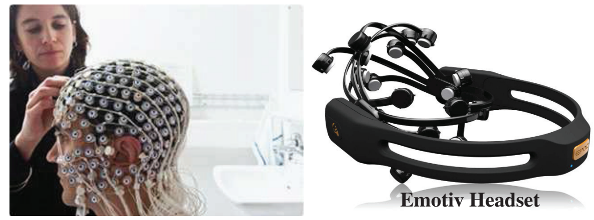

Fig. 1: Experimental (left) versus portable (or wearable) non-invasive BMIs [7].

Among different solutions, wearable systems are the most appealing choice for widespread use in IoT scenarios, since they do not require surgery or expensive equipment. However, the major impediment in the applicability of BMI systems, in particular wearable ones, is the inaccuracy in capturing the user intent. Therefore, *the goal of this paper* is to present a system-level methodology that enables robust communication with IoT devices using inherently unreliable wearable BMI systems. There are efforts to improve the accuracy with better sensors [20] and classification algorithms [16]. However, to the best of our knowledge, this is the first demonstration of using system-level optimization techniques to improve BMI based communication. The proposed methodology consists of two techniques. First, we propose a command confirmation protocol that suppresses errors due to misinterpretation of BMI events. Then, we present an off-line event selection algorithm that extracts an optimal set of events for a given user. The optimality is achieved by user-specific reliability metrics proposed for the first time in this paper. Finally, we experimentally test our methodology in a virtual IoT environment—which resembles an automated assistive home for a disabled person—by carrying out simple and complex tasks using a BMI headset.

The need for design automation: State-of-the-art BMI systems differentiate dozens of mental events (*e.g.*, *push*, *pull*), facial expressions (*e.g.*, *blink*, *smile*), and emotions (*e.g.*, *excitement*, *stress level*) [7]. We formally define the accuracy of an event, such as mental *push*, as the conditional probability of detecting a mental *push* signal given that the user intends to produce a *push*. The accuracy of an event varies widely among users. Moreover, the ability of a given user to generate different events shows large variations. For example, a user may generate the *blink* event very accurately, but face difficulties with cognitive events. Since a user can master a limited number of events, there is a need to select a subset of supported events. The selection problem itself is combinatorial in nature, while assessing the quality of a given subset requires theoretical analysis similar to signal-to-noise

978-1-4673-8388-2/15/$31.00 ©2015 IEEE

ratio computations. Hence, there is a need for optimization algorithms that can identify a robust subset of available events. Finally, errors caused by misinterpreting the events should be suppressed by high level protocols similar to error correction mechanisms. Towards this end, our major contributions are as follows:

- We propose a command confirmation protocol between the user and the receiving device to increase the accuracy of BMI communication,
- We define novel probabilistic metrics to assess the quality of a set of BMI events. Then, we use these metrics for selecting the optimum subset of events for a given user,
- We perform a thorough experimental evaluation using the Emotiv neuroheadset [7] and eight users.

The rest of the paper is organized as follows. Related work is presented in Section II. The confirmation protocol and event selection methodology are explained in Section III and Section IV, respectively. Modelling of compound task is illustrated in Section V and experimental evaluations are presented in Section VI. Finally, conclusions are summarized in Section VII.

II. Related Research

Using BMI for interacting with IoT devices is a promising approach especially for elderly and people with limited motor skills. At the same time, effective use of BMI can revolutionize human interactions with smart networked devices, since the most intuitive and direct source of human intent is the brain itself. Appliance control with BMI for severely paralyzed patients was first proposed about three decades back [23]. Following this path, impressive progress in the use of non-invasive BMI started to appear in the beginning of last decade [18, 30]. Recent achievements include brain-machine-spinal cord interface for function restoration [24], BMI based detection of eye movement intents to assist locked-in patients with immobile eyes [12], brain-controlled functional electrical stimulation of muscles [8], and robotic rehabilitative devices [4, 29]. These applications demonstrate the potential of BMI-based communication. However, training a BMI system is very time consuming and learning new skills afterwards can be exhausting. The authors of [3] address this issue by developing an adaptive hierarchical brain-computer interface architecture which can learn new skills on-the-fly. Introduction of wearable non-invasive BMI systems, such as Emotiv headset [7], have expanded the horizon of BMI research to innovative mobile systems [10, 11, 15, 25].

Despite the impressive progress, BMI systems have reliability issues as highlighted by many researchers [5, 14]. Traditionally, the reliability problem is addressed by using enhanced electrodes [17, 19] and EEG signal classification techniques [21, 27, 28]. However, relying only on the improvements in sensing and processing are insufficient to completely eliminate errors, since the accuracy depends on emotional state of the user [22], environmental noise [31], and electrode placement and condition [22, 31]. These factors are time-variant and nearly impossible to model accurately. Therefore, the major novelty of this paper is introducing a high level protocol and optimum event selection methodology, which boost the robustness of BMI communication.

In contrast to prior work, the approach presented in this paper opens up a new direction to improve BMI system robustness through system-level innovations. In analogy with communication systems, the proposed direction and algorithms deliver reliable operation using error-prone physical channels.

III. Robust BMI Communication Mechanism

A. Overview and Preliminaries

The major problem addressed in this paper is the inaccuracy in interpreting the BMI events. Let $E = \{e_i, 1 \leq i \leq N\}$ be the set of events that can be generated by the target BMI system, where N is the total number of supported events. Suppose that the user tries to generate (*intends*) a mental *push* event. There are three possible outcomes. The BMI system may

- generate correctly a mental *push* event,
- generate another supported event like mental *pull*,
- miss the event (*i.e.*, it may not generate any output).

An event is missed when the classification algorithm cannot match the input EEG signal to any one of the supported events. We denote the conditional probability of inferring e_j given that the user intends to generate e_i as $p(e_j \text{ inferred}|e_i \text{ intended})$, and use shorthand notation $p(j|i)$.

Definition 1. *For any event e_i,*

a) $p_{c,i} = p(i|i)$ *is called the probability of* ***correct*** *interpretation of event e_i.*

b) $p_{f,i} = \sum_{\forall j \neq i} p(j|i)$ *is called the probability of* ***false*** *interpretation of event e_i.*

c) $p_{n,i} = 1 - p_{c,i} - p_{f,i}$ *is called the probability of* missing *the user intent. That is, $p_{n,i}$ is the probability that* ***nothing*** *is detected given that e_i is intended.*

In the following, we first present the command confirmation protocol, since the protocol and definitions therein are an integral part of the event selection algorithm.

B. Command Confirmation Protocol

False interpretation of any event can lead to a hazard, unless the communication with the BMI system is protected using a high level protocol. Therefore, we propose the command confirmation protocol illustrated in Figure 2 to significantly reduce the probability of false interpretation ($p_{f,i}$). Suppose

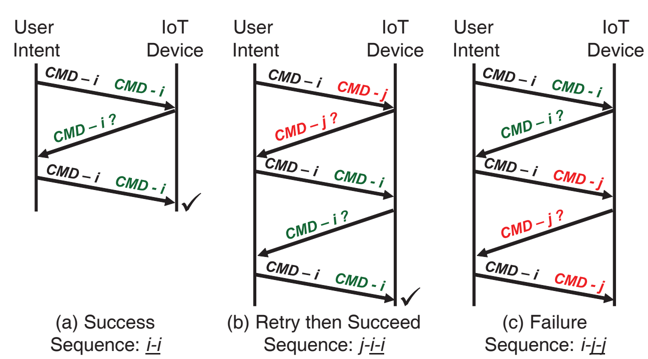

Fig. 2: Event diagrams for success, retry, and failure outcomes.

that the user wants to control an IoT device using one of the four available commands $\{ON, OFF, UP, DN\}$, where *UP* (*DN*) is used for increasing (decreasing) the volume of the device. Upon receiving a command, the IoT device echoes the inferred command by sending a visual feedback such as blinking lights or turning on a particular LED[1], rather than taking an immediate action. The user needs to confirm the command, *i.e.*, the device needs to receive the same command twice consecutively before taking an action.

Unlike automatic repeat request (ARQ) protocols [2], which typically rely on a perfect acknowledgement signal, there is no guarantee that the confirmation will match the original intent. The user intent can be captured accurately, only if both the original command and confirmation are inferred successfully, as illustrated in Figure 2. When the original command and confirmation are interpreted differently, the IoT device will continue asking for confirmation until inferring the same command twice. There is a non-zero probability that the original intent is mistaken with the same event twice consecutively, as depicted in Figure 2c. In this case, the IoT device will take an unintended action albeit with a smaller probability. Suppose that we want to turn the IoT device off in our example. The sequence *OFF-DN-OFF-OFF* would turn the device off in four steps, while the sequence *OFF-DN-OFF-DN-DN* would turn the volume down *by mistake*. Assuming that the probability of correct interpretation is larger than the probability of false interpretation ($p_{c,OFF} > p_{f,OFF}$), the latter sequence would occur with much smaller probability, as shown by the analysis in Section III-C. Finally, the events can also be missed with probability $p_{n,i}$. In this case, the receiving IoT device will not take any action.

The probability of failure can be decreased further by increasing the number of confirmations. However, we observed that the improvement in probability of success was negligible beyond what was achievable with a single confirmation, while the expected number of trials increased significantly. We could have also used a separate event (*e.g.*, the most reliable event) as the confirmation. We discarded this option as we observed that users are more comfortable repeating the same command rather than alternating between two events. Moreover, using the same event for confirmation enables the user to repeat the command without necessarily waiting for the echo from the IoT device. Finally, we note that it is also possible to add a time-out that would re-initialize the system to bound the number of retries.

C. Probability of Success and Expected Time Analysis

We modeled the proposed command confirmation protocol using a discrete-time Markov chain (DTMC), as shown in Figure 3. Suppose that the user intention is event e_i. From the *Initial* state, the receiving IoT device enters the *Confirmation* state (correct interpretation) with probability $p_{c,i} = p(i|i)$. Likewise, it can enter the *Retry* state (false interpretation) with probability $p_{f,i} = \sum_{\forall j \neq i} p(j|i)$, or stay in the *Initial* state with probability $p_{n,i} = 1 - p_{c,1} - p_{f,i}$ (missed event). When the

[1]In our experiments, we displayed the inferred command on a monitor screen as a visual feedback to the user.

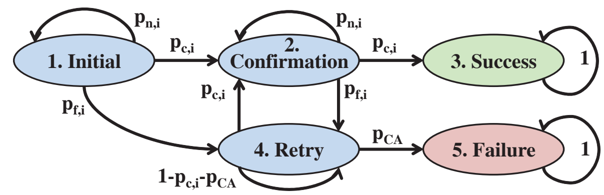

Fig. 3: The DTMC model of the command confirmation protocol.

device is in the *Confirmation* state, it enters the *Success* state upon one more correct interpretation, *i.e.*, with probability $p_{c,i} = p(i|i)$. However, the device moves to *Retry* state in case of a false interpretation, and stays in the *Confirmation* state if the event is missed, as depicted in Figure 3. Likewise, when it is in the *Retry* state, a correct interpretation can bring the system to the *Confirmation* state with probability $p_{c,i}$. From the *Retry* state, the device can also move to the *Failure* state, as a result of two consecutive false interpretations of any unintended event. This probability is denoted by p_{CA}, and computed as follows:

$$p_{CA} = \sum_{\forall j \neq i} p(j|i).p(j|i) = \sum_{\forall j \neq i} p^2(j|i) \quad (1)$$

The initial probability distribution is $\boldsymbol{\pi_{0,i}} = [1\ 0\ 0\ 0\ 0]$, since we always start at the *Initial* state, which is state "1" in Figure 3. The probability distribution at time step $k+1,\ k \geq 0$ can be obtained iteratively using $\boldsymbol{\pi_{k+1,i}} = \boldsymbol{\pi_{k,i}} \times \boldsymbol{P_i}$, where $\boldsymbol{P_i}$ is the state-transition matrix:

$$\boldsymbol{P_i} = \begin{bmatrix} p_{n,i} & p_{c,i} & 0 & p_{f,i} & 0 \\ 0 & p_{n,i} & p_{c,i} & p_{f,i} & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & p_{c,i} & 0 & 1 - p_{c,i} - p_{CA} & p_{CA} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

The key metric of interest is the probability of reaching the *Success* state, as defined next.

Definition 2. *For any event e_i, the probability of reaching the absorbing Success state is defined as the probability of success, $P_{success}$. Since* Success *and* Failure *are the only two absorbing states, the probability of failure is given by* $P_{failure} = 1 - P_{success}$.

The probability of success is a function of the probability of correct interpretation ($p_{c,i}$), false interpretation ($p_{f,i}$), and missing an event ($p_{n,i}$), which are given in Definition 1. Figure 4a shows the probability of success for event e_i in a BMI system with four events. These results are obtained by sweeping $p_{c,i}$ and $p_{n,i}$, while equally distributing $p_{f,i}$ to the rest of the commands. For a given $p_{n,i}$, the probability of success increases as $p_{c,i}$ increases. For example, when $p_{n,i} = 0$ (■ marker in Figure 4a), the proposed protocol delivers probability of success $P_{success} = 0.95$ even with a modest probability of correct interpretation of $p_{c,i} = 0.65$. We also observe that the probability of success increases with increasing $p_{n,i}$ for a given $p_{c,i}$, as illustrated with different markers in Figure 4a. In particular, we can achieve $P_{success} > 0.95$ with $p_{c,i}$ as low as 0.45, when the probability

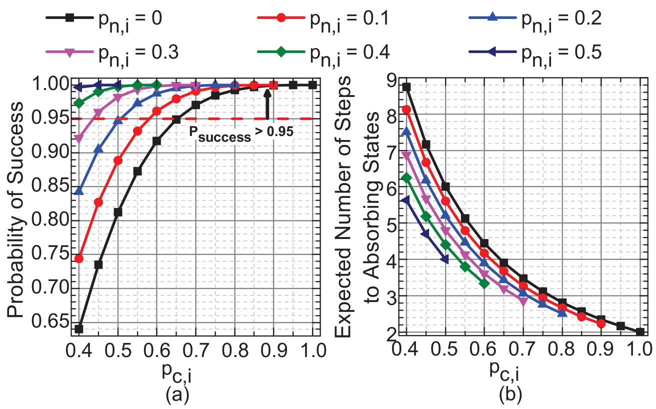

Fig. 4: (a) The probability of success ($P_{success}$) and (b) expected number of steps achieved by the proposed command confirmation protocol.

of missing an event $p_{n,i} = 0.3$. Larger $p_{n,i}$ improves the probability of success, since it implies smaller probability of false interpretation $p_{f,i}$ for a given $p_{c,i}$. In other words, missing the event more often results in smaller number of false interpretations and decreases the probability of failure.

Figure 4b shows that lower number of false interpretations for a given $p_{c,i}$ also reduces the expected number of steps to reach the absorbing states. This is intuitive since no move is better than a false move. Plots with different markers show that larger $p_{c,i}$ and $p_{n,i}$ (*i.e.*, smaller $p_{f,i}$) reduce the time to reach the absorption states. Finally, as the probability of correct interpretation $p_{c,i}$ approaches to 1.0, the expected number of steps decreases and converges to 2. This is expected since the minimum number of steps to reach the absorbing states *Success* or *Failure* state from the *Initial* state is 2, as illustrated in Figure 3. Increasing number of steps has a small impact on user experience, since commands can be repeated in the fraction of a second.

IV. Event Selection Methodology

In general, BMI systems can generate a large number of events. For example, the experimental headset used in this work can generate thirty one events. However, a user can master and manage only a subset of these events. Thus, we present a methodology to select the optimum set of K out of N supported events $\mathcal{S}_K \subset E$ for a particular user. Our methodology consists of the quality assessment of a given selection, and a branch-and-bound algorithm that searches the event space to find the best set of K events.

A. Systematic Training and Quality Assessment

The first step in identifying the optimum subset of events for a given user is to train the user for as many different events as possible. After completing the training, we generate an $N \times N$ event probability matrix, $\boldsymbol{M_N}$, which describes the accuracy of interpreting each supported event. More precisely,

$$\boldsymbol{M_N} = \begin{bmatrix} p(1|1) & p(1|2) & \cdots & p(1|N) \\ p(2|1) & p(2|2) & \cdots & p(2|N) \\ \vdots & \vdots & \vdots & \vdots \\ p(N|1) & p(N|2) & \cdots & p(N|N) \end{bmatrix} \tag{3}$$

The diagonal entries of $\boldsymbol{M_N}$ give the probability of correct interpretation of each event, *i.e.*, $p_{c,i} = p(i|i)$. Likewise, the i^{th} column lists conditional probabilities of inferring each event given that the user intention is event e_i. We note that large values at the diagonal are favourable but not sufficient to select an event over other available events. It is also crucial to ensure that the probability of confusing an event with the *other events in the selected set* is small. Therefore, we need a quality metric for the subset $\mathcal{S}_K$ to encapsulate both the probability of success and expected number of trials. This requirement is formalized by the following definitions.

Definition 3. *The conditional probability $p(j|i)$ is called the* aversion *of event i towards event j, $j \neq i$.*

That is, *aversion* measures the probability of inferring another event given that the user *intents* event e_i. Referring to Figure 4a, we can achieve higher probability of success when $p_{n,i}$ increases, which implies smaller *aversion probability* for a given $p_{c,i}$. Using the *aversion* definition, we can express the strength of event e_i for subset $\mathcal{S}_K$ as:

$$\text{Strength}_{i,\mathcal{S}_K} = \frac{p^2(i|i)}{\sum_{j \subset \mathcal{S}_K, j \neq i} p^2(j|i)} \tag{4}$$

With respect to the DTMC shown in Figure 3, the event strength is the ratio of two probabilities: the probability of reaching the *Success* state over the probability of reaching the *Failure* state in minimum number of steps.

Definition 4. *The conditional probability $p(i|j)$ is called the* affinity *of event j, $j \neq i$ towards event i.*

That is, *affinity* measures the probability of spuriously inferring event e_i when the user intends another event. Similar to *aversion*, small *affinity* with other events is desirable. The affinity of event e_i for subset $\mathcal{S}_K$ can be represented as:

$$\text{Affinity}_{i,\mathcal{S}_K} = \sum_{j \subset \mathcal{S}_K, j \neq i} p(i|j) \tag{5}$$

That is, the affinity of e_i for a *given subset* $\mathcal{S}_K$ is the sum of probabilities in the i^{th} row of the event probability matrix. We also need to consider the expected number of steps required to enter the absorbing states $E[T_i]$ to avoid exploding the response time, as depicted in Figure 4b. Therefore, we define the quality of event e_i for a subset $\mathcal{S}_K$ using equation 4 and 5:

$$\text{Quality}_{i,\mathcal{S}_K} = \frac{\text{Strength}_{i,\mathcal{S}_K}}{\text{Affinity}_{i,\mathcal{S}_K} \times E[T_i]} \tag{6}$$

Note that $\text{Quality}_{i,\mathcal{S}_K}$ is computed for each event in a given subset $\mathcal{S}_K$. Therefore, we employ $L1$ norm to obtain the overall quality metric for $\mathcal{S}_K$,

$$\|\text{Quality}_{\mathcal{S}_K}\|_1 = \sum_{i \subset \mathcal{S}_K} |\text{Quality}_{i,\mathcal{S}_K}| \tag{7}$$

Finally, the optimal solution is the subset $\mathcal{S}_K$ with largest $\|\text{Quality}_{\mathcal{S}_K}\|_1$ among all possible subsets of $E = \{e_i, 1 \leq i \leq N\}$, *i.e.*, the set of supported events. Next, we present an algorithm to find the optimal subset $\mathcal{S}_K$.

B. Branch-and-Bound Algorithm

The number of all possible combinations of choosing a subset K from N events is $\binom{N}{K}$. A brute-force technique for finding $\mathcal{S}_K$ becomes inefficient as the problem-size grows, which is expected as the wearable BMI systems become more advanced. Therefore, we model the search space by a search tree, where the root is an empty list. The leaves represent complete solutions, and the intermediate nodes are partial solutions with less than K events. A branch-and-bound algorithm is effective since we can utilize Equation 7 for both evaluating each complete solution and finding an upper bound for each intermediate node.

Branch: To find the optimal solution, we perform a depth-first-search. When a leaf node is reached, we use Equation 7 to compute the corresponding quality metric. If it results in a better quality solution than the maximum quality found to that point, we update the optimum solution. Then, the search traces back to the next unexpanded intermediate node and continues until traversing the whole tree.

Bound: When the search traces back to one of the intermediate nodes or the root, we compute an upper bound for the quality that can be achieved by expanding the current solution. If the upper bound is smaller than the maximum quality found to that point, then we skip that node and continue tracing back. Since each node, except for the leaves, represents an incomplete list, we find the upper bound by padding the list with ideal events for which $p(i|i) = 1$, and $p(j|i) = 0,\ \forall j \neq i$. As the problem size grows, the bound can be made tighter by using the best set of events that are not already selected.

The run time of the algorithm is in the order of a couple of minutes when working with 31 events on an Intel® Xeon® processor E5-2630 v2 and 32 GB RAM. Our experiments show on average $4.42\times$ speed-up compared to a brute-force solution even when selecting 4 out of 31 events.

V. Modeling Compound Tasks

We categorize the tasks that require a sequence of commands for successful completion as compound tasks. For instance, browsing in a graphical user interface to select a menu item or 2-dimensional (2-D) navigation using *forward*, *right*, *left*, and *stop* commands can be considered as compound tasks. A compound task requires separate BMI events to perform each simple task. The success of the overall compound task depends on the probability of correct ($p_{success}$) and false interpretation ($p_{failure}$) of individual BMI events being used. At the same time, the user can counteract a false interpretation to correct the course of operation.

To illustrate the modeling of a compound task using DTMC, we consider 2-D navigation of a wheelchair using four simple commands, *forward*, *right*, *left*, and *stop*. The model has five states, namely *At Rest*, *Turning Left*, *Turning Right*, *Going Forward*, and *At Target*, as shown in Figure 5. Initially, the wheelchair is assumed to be in the *At Rest* state with an arbitrary orientation.

Ideal (error-free) Operation: If the wheelchair is not directly facing the target, the system moves to *Turning Left* or *Turing Right* such that it can align with the target. If the wheelchair

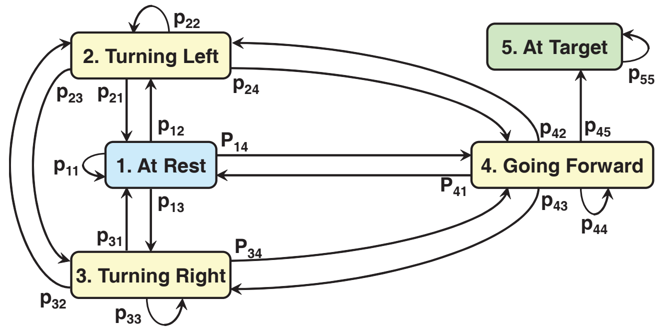

Fig. 5: The DTMC model of compound navigation task.

faces the target, it moves to the *Going Forward* state. The wheelchair moves until reaches the target and enters the *At Target* state.

Actual Operation: Due to the non-zero failure probability of BMI communication, the wheelchair can make an unintended move. For example, suppose that the wheelchair is in the *At Rest* state and facing the target. Naturally, the user would send a *forward* command such that the wheelchair would enter the *Going Forward* state. If the *forward* command is misinterpreted, the wheelchair can start rotating left or right. This corresponds to entering the absorbing state *Failure* in Figure 3 with probability $P_{failure}$. Hence, the user need to react to correct the course of action.

The state transition probabilities p_{ij} in Figure 5 denote the probability of moving from state i to state j. These probabilities are a function of the accuracy of conveying individual BMI events and plotted in Figure 4(a). As the probability of correct interpretation $p_{c,i}$ approaches to 1, the randomness reduces, and the expected time to reach the target decreases, since we can achieve the ideal operation. In the other extreme, the wheelchair would make a 2-D random walk, if the interpretations were random. In other words, using the proposed protocol and event selection algorithm increase $p_{c,i}$, which make the completion of compound tasks faster.

VI. Experimental Evaluation

A. Experimental Setup and Methodology

Our experimental setup consists of a wearable BMI system from Emotiv [7], and a virtual robot experimentation platform (V-REP) [9]. The Emotiv headset, shown in Figure 1, collects raw EEG data using fourteen electrodes, and transmits encrypted EEG signals to a software development kit (SDK). The SDK can classify a total of thirty one events including facial expressions, subjective emotional responses, and conscious intents. Then, it converts the user intentions into keystrokes. We developed a framework where these keystrokes can be used as commands to control IoT devices in a virtual home environment constructed in V-REP.

Methodology: The Emotiv headset, like other BMI systems, requires training for effective use. Users train for each supported event, such as mental *push* and *pull*, separately. During the training for a specific event e_i, the SDK stores the raw EEG signals. Then, the classification algorithms use these stored EEG signals at runtime to recognize the user intent.

In our experiments, we worked with a total of eight users, and employed four BMI events which were sufficient to accomplish a complex navigation task in a 2-D environment. The users trained for each supported event following the guidelines provided by the Emotiv SDK for at least three hours. A particular event training was considered complete when the user was satisfied with the performance, or chose to stop due to hardship in generating the event. As the baseline, each user first picked the four most effective events they were comfortable with. Then, we mapped the selected events to different commands, for instance, *blink* event to *stop* command and mental *push* to *forward*. Then, we applied the methodology presented in Section IV in two phases. First, we generated the event probability matrix $\boldsymbol{M_N}$ defined in Equation 3 for each user. This is accomplished by having the users try each event fifty times, and finding the conditional probabilities. In the second phase, we applied the branch-and-bound algorithm to select the optimum subset of events $\mathcal{S}_4$ using the $\boldsymbol{M_N}$ matrix, and called it as the User–Skill matrix:

$$\text{User}-\text{Skill}_4 = \begin{bmatrix} p(1|1) & p(1|2) & p(1|3) & p(1|4) \\ p(2|1) & p(2|2) & p(2|3) & p(2|4) \\ p(3|1) & p(3|2) & p(3|3) & p(3|4) \\ p(4|1) & p(4|2) & p(4|3) & p(4|4) \\ p(\phi|1) & p(\phi|2) & p(\phi|3) & p(\phi|4) \end{bmatrix} \tag{8}$$

The last row gives the probability that the user intent is e_i, but nothing (ϕ) is inferred. This could happen if the event is missed or the EEG signals cannot be matched to any one of the selected events in the subset.

Next, we analyze the accuracy of performing simple, compound, and complex tasks under three scenarios:

- ***Ad hoc***: Users selected the events based on perceived comfort level, and *did not adopt* the command confirmation protocol.
- ***Event selection***: Users followed the proposed event selection methodology to select the BMI events, but they *did not adopt* the command confirmation protocol.
- ***With protocol***: Users followed the proposed event selection methodology and the command confirmation protocol.

B. Simple Tasks

Controlling of IoT home appliances like lights, fans, and other actuators can be categorized as simple tasks. For example, a light bulb can be turned on or off using only one command. Thus, the success of completing the task is the same as the success of a single event. Figure 6 shows the accuracy in conveying a single command for the three scenarios described in Section VI-A. We observed that the probability of correct interpretation $p(i|i)$ varied between 0.52–0.86 before applying the event selection algorithm. The event selection algorithm alone increased the range of probability of correct interpretation to 0.68–0.94. Using the command confirmation protocol further increased the probability of success above 0.95 for 7 out of 8 users, and to 0.90 for User-1. In particular, the accuracy for User-7, who experienced the lowest accuracy after event selection, increased from 0.68 to 0.98. Although the

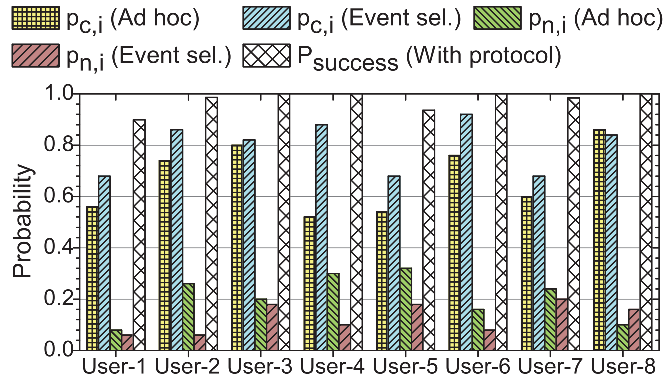

Fig. 6: Comparison of probability of success for eight users with and without using the approach in the paper.

protocol requires two successful event generations instead of one, it reduces overall task completion time by eliminating the time needed to correct the aftermath of a false event generation. Hence, actuation of simple IoT devices is possible with very high accuracy by using the approach presented in this paper. Finally, we note that impact of the algorithm on the events with lower accuracies is more pronounced. Their plots are omitted due to space considerations.

C. Compound Task

Compound tasks consist of a series of simple tasks, as explained in Section V. To evaluate the performance of the proposed approach in completing compound tasks, we chose navigation from one point to another. Figure 7 shows the experimental layout in a virtual room constructed in V-REP. We placed a wheelchair at point A, and set point B as the target such that the wheelchair was not facing the target. In order to complete the task, the wheelchair should make a state transition to *Turning Left* state, then move to *Going Forward* state when the wheelchair is facing the target. Finally, it should enter the *At Target* state upon reaching point B.

We first analyzed the DTMC model to calculate the number of steps required to reach the absorbing state *At Target*. Figure 8 shows that the event selection algorithm alone reduces the number of steps required to complete the task by 54% on average. Our command confirmation protocol on top of event selection demonstrated further improvements. More precisely, the command confirmation protocol delivered on average 79.7% reduction compared to ad hoc BMI event selection. Overall, it is evident that the improvements in the accuracy of individual commands result in even larger benefits when executing compound tasks.

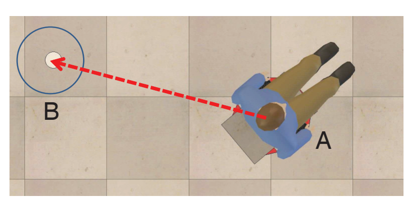

Fig. 7: Layout of compound navigation task.

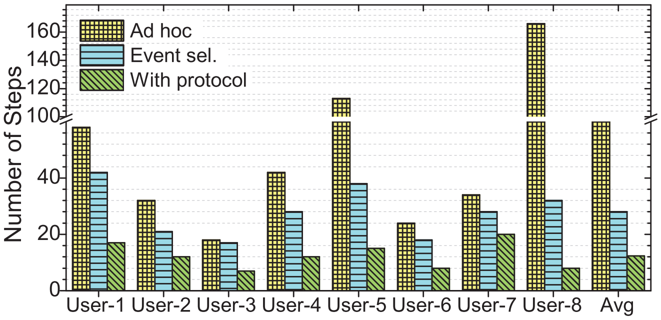

Fig. 8: Expected number of steps to complete compound navigation task using ad hoc selected events, events selected using described methodology, and with command confirmation protocol.

We also performed Monte Carlo simulations by emulating the users in Matlab to assess the impact of the proposed approach under a large number of trials. To emulate the users, we used the event probability matrix $M_{\boldsymbol{N}}$, the subset $\mathcal{S}_4$ given in Equation 8, and the results of simple experiments. Matlab generated the user intent based on the position and orientation of the wheelchair. Then, the intents were passed to V-REP, where the physical movements were modeled. Finally, the updated physical data such as position and orientation was fed back from V-REP to Matlab. For each of the eight users, we performed 100 simulations using commands selected in ad hoc manner, using commands chosen by the event selection methodology, and adapting the command confirmation protocol. That is, a total of 2400 different simulations were performed.

Figure 9 shows that completion time for experiments using ad hoc approach varied between 15s (only for five cases) and 125s. Using the proposed event selection algorithm not only pushed the histogram to left, but also decreased the variation. For instance, the maximum completion time reduced to 65s compared to 125s obtained using ad hoc selection. Finally, adding the command confirmation protocol on top of the event selection algorithm resulted in significant improvements both in the average and maximum time to reach the destination. In particular, only four users took more than 40s to reach the target, while the average time to reach the target was reduced to 21.5s, which implies 63% improvement compared to the ad hoc choice.

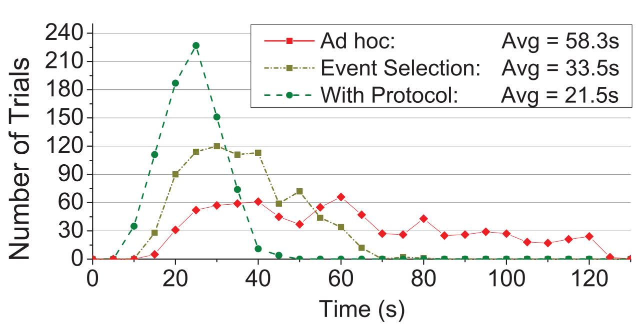

Fig. 9: Simulation result for compound navigation task.

D. Complex Mission

Tasks, such as navigation in a room full of obstacles, or picking up objects by controlling prosthetic limbs are called complex tasks, since they require a sequence of compound tasks to complete. A possible complex task scenario is a robot that assists people with disabilities with the high level commands from the user in an environment populated with IoT devices [6]. For example, the robot can align itself with the door frame to pass through the door safely, and locate an object in the house with the help of the smart sensors with minimum human intervention.

Scenario: To emulate a realistic scenario, we modeled the virtual home environment shown in Figure 10. Users generated *forward*, *stop*, *left*, and *right* commands using the Emotiv headset to navigate the wheelchair from the starting point to the target while visiting milestones-1–3, which represented exiting the first room, drinking water from the table, entering the second room, and meeting a person.

Implementation: The BMI commands from the users are converted to keystrokes in real-time and fed to Matlab. In experiments without the protocol, a Matlab script generates the low level control signals for each wheel, and sends them to V-REP through an API. When applying the protocol, the Matlab script also prints the keystroke on the command line and waits for confirmation, as illustrated in Figure 2. In case of a false interpretation, *e.g.*, turning left instead of stopping, the user is able to respond in real-time to correct the route.

Experiment: Seven out of eight trained users participated in this experiment. They used the BMI headset to navigate the disabled person with and without the proposed approach. The completion time for users following the ad hoc event selection ranged from 107.8s – 277.2s and averaged 179.2s, as summarized in Figure 11. The completion times relied heavily on the user skill, and had standard deviation of 54.0s. We also observed occasional crashes for several users.

The proposed approach resulted in 124.9s average completion time, which means almost $1.5\times$ improvement over the ad hoc BMI communication. We note that User-4 was an outlier with a completion time of 243.0s, while the completion times of the rest varied between ranged from 89.7s – 128.2s, with a standard deviation of 15.8s. Finally, we observed only one crash (User-1) when using the proposed methodology.

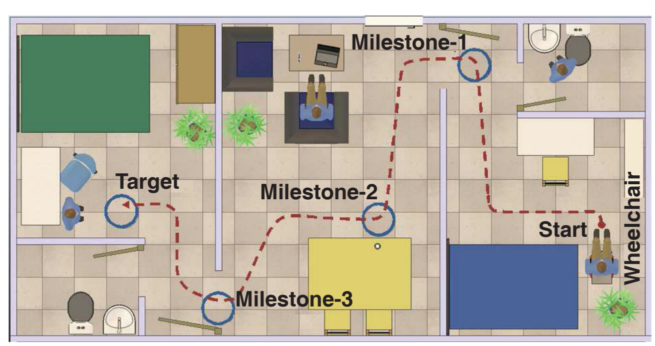

Fig. 10: Layout of the complex mission in V-REP.

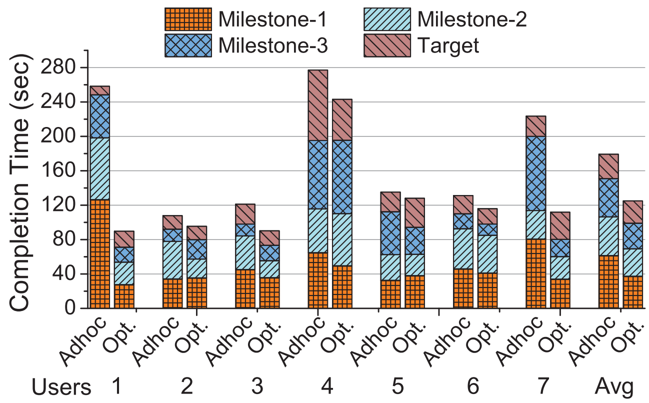

Fig. 11: Completion times for milestones 1-4 for seven users in complex navigation mission.

We emphasize that navigating the wheelchair by *only* using BMI inputs is a very challenging task. Having seven users perform this task with and without the proposed approach was a major effort. As a reference point and preparation to this experiment, we also asked the users to perform the same task using the key strokes like playing a game. The completion times were consistently around 70s. This means that users 1 – 3 were within 20% of the ideal completion time, while users 5 – 7 took about 50% longer.

VII. Conclusion

Non-invasive wearable BMIs are promising candidates for interacting with the IoT ecosystem. Besides helping healthy individuals, wearable BMIs can be a key enabler for people suffering from disabilities, neurodegenerative disorders or limited motor skills due to aging. This paper showed that it is possible to alleviate inherent inaccuracies in state-of-the-art BMI systems using system-level protocols and design automation algorithms. Our experiments using a state-of-the-art BMI headset demonstrated that complex tasks can be performed accurately using BMI. More precisely, we showed on average 63% improvement over an ad hoc approach in the completion of a 2-D navigation example. These results open up a new direction to improve BMI system robustness through innovations at higher abstraction layers.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A Survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] D. P. Bertsekas, R. G. Gallager, and P. Humblet, *Data Networks*. Prentice-hall Englewood Cliffs, NJ, 1987, vol. 2.

[3] M. Bryan, G. Nicoll, V. Thomas, M. Chung, J. R. Smith, and R. P. Rao, "Automatic Extraction of Command Hierarchies for Adaptive Brain-Robot Interfacing," in *Proc. of Intl. Conf. on Robotics and Automation*, 2012, pp. 3691–3697.

[4] U. Chaudhary, N. Birbaumer, and M. Curado, "Brain-Machine Interfaces (BMI) in Paralysis," *Annals of Physical and Rehabilitation Medicine*, vol. 58, no. 1, pp. 9–13, 2015.

[5] F. Cincotti *et al.*, "Non-invasive Brain–Computer Interface System: Towards its Application as Assistive Technology," *Brain Research Bulletin*, vol. 75, no. 6, pp. 796–803, 2008.

[6] M. C. Domingo, "An Overview of the Internet of Things for People with Disabilities," *Jrnl. of Net. and Comp. App.*, vol. 35-2, pp. 584–596, 2012.

[7] Emotiv. EEG neuroheadset. http://emotiv.com/store/hardware/epoc-bci-eeg/developer-neuroheadset/.

[8] C. Ethier and L. E. Miller, "Brain-Controlled Muscle Stimulation for the Restoration of Motor Function," *Neurobiology of Disease*, 2014.

[9] M. Freese, S. Singh, F. Ozaki, and N. Matsuhira, "Virtual Robot Experimentation Platform V-REP: A Versatile 3d Robot Simulator," in *Sim., Modeling, and Prog. for Auto. Robots*, 2010, pp. 51–62.

[10] K. George, A. Iniguez, H. Donze, and S. Kizhakkumthala, "Design, Implementation and Evaluation of a Brain-Computer Interface Controlled Mechanical Arm for Rehabilitation," in *Proc. of Intl. Instrumentation and Measurement Technol. Conf.*, 2014, pp. 1326–1328.

[11] F. Gong, W. Xu, J.-Y. Lee, L. He, and M. Sarrafzadeh, "NeuroGlasses: A Neural Sensing Healthcare System for 3-D Vision Technology," *IEEE Trans. on Inf. Technol. in Biomedicine*, vol. 16, no. 2, pp. 198–204, 2012.

[12] A. B. Graf and R. A. Andersen, "Brain–Machine Interface for Eye Movements," *Proc. of the National Academy of Sci.*, vol. 111, no. 49, pp. 17 630–17 635, 2014.

[13] I. S. Jose M. Carmena. How to Control a Prosthesis With Your Mind. Http://spectrum.ieee.org/biomedical/bionics/how-to-control-a-prosthesis-with-your-mind.

[14] M. A. Lebedev and M. A. Nicolelis, "Brain–Machine Interfaces: Past, Present and Future," *TRENDS in Neurosciences*, vol. 29, no. 9, pp. 536–546, 2006.

[15] W. T. Lee, H. Nisar, A. S. Malik, and K. H. Yea, "A Brain Computer Interface for Smart Home Control," in *IEEE International Symposium on Consumer Electron. (ISCE)*, 2013, pp. 35–36.

[16] Y.-H. Liu, H.-P. Huang, T.-H. Huang, Z.-H. Kang, and J.-T. Teng, "Controlling a Rehabilitation Robot with Brain-Machine Interface: An approach based on Independent Component Analysis and Multiple Kernel Learning," *Intl. Jrnl. of Automation and Smart Tech.*, vol. 3, no. 1, pp. 67–75, 2013.

[17] H. S. Mandal *et al.*, "Improving the Performance of Poly (3, 4-ethylenedioxythiophene) for Brain–Machine Interface Applications," *Acta Biomaterialia*, vol. 10, no. 6, pp. 2446–2454, 2014.

[18] J. d. R. Millán, F. Renkens, J. Mourino, and W. Gerstner, "Non-Invasive Brain-Actuated Control of a Mobile Robot," in *Proc. of Intl. Joint Conf. on Artificial Intelligence*, 2003, pp. 1121–1126.

[19] S. Morishita *et al.*, "Brain-Machine Interface to Control a Prosthetic Arm with Monkey ECoGs during Periodic Movements," *Frontiers in Neuroscience*, vol. 8, pp. 1–9, 2014.

[20] K. A. Moxon *et al.*, "Nanostructured Surface Modification of Ceramic-Based Microelectrodes to Enhance Biocompatibility for a Direct Brain-Machine Interface," *IEEE Trans. on Biomed. Eng.*, vol. 51, no. 6, pp. 881–889, 2004.

[21] J. Müller-Gerking, G. Pfurtscheller, and H. Flyvbjerg, "Designing Optimal Spatial Filters for Single-Trial EEG Classification in a Movement Task," *Clinical Neurophysiology*, vol. 110, no. 5, pp. 787–798, 1999.

[22] F. Nijboer, N. Birbaumer, and A. Kübler, "The Influence of Psychological State and Motivation on Brain–Computer Interface Performance in Patients with Amyotrophic Lateral Sclerosis–A Longitudinal Study," *Frontiers in Neuroscience*, vol. 4, pp. 1–13, 2010.

[23] E. M. Schmidt, "Single Neuron Recording from Motor Cortex as a Possible Source of Signals for Control of External Devices," *Annals of Biomed. Eng.*, vol. 8, pp. 339–349, 1980.

[24] S. Shahdoost *et al.*, "Towards a Miniaturized Brain-Machine-Spinal Cord Interface (BMSI) for Restoration of Function After Spinal Cord Injury," in *Eng. in Medicine and Biol. Soc. (EMBC)*, 2014, pp. 486–489.

[25] P. Simoens and other, "Vision: Smart Home Control with Head-Mounted Sensors for Vision and Brain Activity," in *Proc. of the fifth International Workshop on Mobile Cloud Comput. & Services*, 2014, pp. 29–33.

[26] J. Stankovic, "Research Directions for the Internet of Things," *Internet of Things Jrnl., IEEE*, vol. 1, no. 1, pp. 3–9, 2014.

[27] H.-I. Suk and S.-W. Lee, "A Novel Bayesian Framework for Discriminative Feature Extraction in Brain-Computer Interfaces," *IEEE Trans. on Pattern Anal. and M/C Intelligence*, vol. 35, no. 2, pp. 286–299, 2013.

[28] N. Tomida, M. Yamagishi, I. Yamada, and T. Tanaka, "A Reduced Rank Approach for Covariance Matrix Estimation in EEG Signal Classification," in *Proc. of Intl. Conf. of Engg. in Med. and Bio. Soc.*, 2014, pp. 668–671.

[29] A. Venkatakrishnan, G. E. Francisco, and J. L. Contreras-Vidal, "Applications of Brain–Machine Interface Systems in Stroke Recovery and Rehabilitation," *Current physical medicine and rehabilitation reports*, vol. 2, no. 2, pp. 93–105, 2014.

[30] J. R. Wolpaw and D. J. McFarland, "Control of a Two-Dimensional Movement Signal by a Noninvasive Brain-Computer Interface in Humans," *Proc. of the National Academy of Sci. of the United States of America*, vol. 101, no. 51, pp. 17 849–17 854, 2004.

[31] J. R. Wolpaw *et al.*, "Brain-Computer Interface Technology: A Review of the First International Meeting," *IEEE Trans. on Rehabilitation Engg.*, vol. 8, no. 2, pp. 164–173, 2000.