


Probabilistic simulation framework for EEG-based BCI design

Umut Orhan^a, Hooman Nezamfar^b, Murat Akcakaya^c, Deniz Erdogmus^b, Matt Higger^b, Mohammad Moghadamfalahi^b , Andrew Fowler^d, Brian Roark^e, Barry Oken^d and Melanie Fried-Oken^d

^aHoneywell International Inc., Morris Plains, NJ, USA; ^bNortheastern University, Boston, MA, USA; ^cUniversity of Pittsburgh, Pittsburgh, PA, USA; ^dOregon Health and Science University, Portland, OR, USA; ^eGoogle Inc., New York, NY, USA

ABSTRACT

A simulation framework could decrease the burden of attending long and tiring experimental sessions on the potential users of brain-computer interface (BCI) systems. Specifically during the initial design of a BCI, a simulation framework that could replicate the operational performance of the system would be a useful tool for designers to make design choices. In this manuscript, we develop a Monte Carlo-based probabilistic simulation framework for electroencephalography (EEG) based BCI design. We employ one event-related potential (ERP) based typing and one steady-state evoked potential (SSVEP) based control interface as testbeds. We compare the results of simulations with real-time experiments. Even though over- and underestimation of the performance is possible, the statistical results over the Monte Carlo simulations show that the developed framework generally provides a good approximation of the real-time system performance.

ARTICLE HISTORY

Received 24 June 2015
Accepted 21 October 2016

KEYWORDS

Electroencephalography;
event-related potentials;
steady-state visually evoked
potentials; simulation

1. Introduction

Brain-computer interfaces (BCI) are seen as the future of human-computer interaction. In particular, BCI-enabled devices that will allow people with severe speech and physical impairments (SSPI) to communicate and interact with their personal networks and environments have received significant interest from the research community in the last decades.[1,2] Noninvasive BCI design paradigms, particularly those using electroencephalography (EEG), are increasingly popular due to their portability, noninvasiveness, cost-effectiveness, and reliability.[1–3]

Developing and testing EEG-based BCI systems require extensive experimental analysis which in turn depends on the participation of the BCI users in multiple experimental sessions. This may be not only expensive and time-consuming but also very uncomfortable especially for people with SSPI. Our running hypothesis is that BCI research would benefit from a simulation framework that could possibly eliminate the requirement of extensive experimental studies, especially for the validation of simple system changes, specifically in the early stages of the design. In this manuscript, we describe a probabilistic simulation framework as a tool for EEG-based BCI design. After an initial experiment, this framework has the capability of using the collected data from that experiment to generate

EEG responses in a probabilistic manner to simulate the operation of a BCI system. We validate the simulation framework using two different BCI testbeds: (1) an event-related potential (ERP) based typing BCI testbed that uses rapid serial visual presentation (RSVP) to evoke responses in the EEG of a user (RSVP KeyboardTM) and (2) a steady-state visual evoked potential (SSVEP) based control BCI testbed. In these two testbeds, we conduct experiments and collect EEG data under different system specifications. These experiments consist of a calibration phase and a testing phase. In the calibration phase, EEG data collected in a supervised fashion are used to develop a statistical inference engine that is later used in the test phase to operate the BCI. The proposed probabilistic framework uses the data from the calibration phase of these experiments to simulate the operation in the testing phase. We compare the performance analysis results of the experimental studies with the simulations.

2. Simulation framework

In the proposed simulation framework, we consider order-1 Markov processes with finite dimensional state vectors as shown in Figure 1. This graphical model represents the generative model of the collected data in epoch

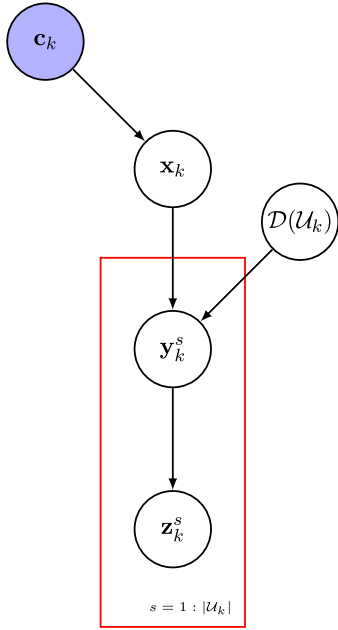


Figure 1. Graphical model (epoch k).

k . In this figure, we have the following variables of interest (at epoch k): (1) \mathbf{x}_k is the latent dynamic state variable that needs to be inferred from the measurements and context information. The state is discrete-valued, taking values from a countable set \mathcal{X} which includes the message (e.g. a symbol in typing or a command in device control) that the BCI user is intending to communicate; (2) \mathbf{c}_k is the context information such as the effect of a language model for a typing interface or a probabilistic map for a BCI-based robot navigation; (3) \mathcal{D} is the dictionary (set) of stimuli that are presented to the BCI user to obtain EEG measurements dependent on \mathbf{x}_k ; (4) \mathcal{U}_k is the index set of stimuli to be presented during the k th epoch; (5) $\mathcal{Y}_k = \{\mathbf{y}_k^1, \dots, \mathbf{y}_k^{|\mathcal{U}_k|}\}$ are the true labels for the presented stimuli set $\mathcal{D}(\mathcal{U}_k)$ given the state such that these true labels take values from a countable set \mathcal{Y} where there exists a deterministic label mapping $\mathcal{L}: \mathcal{X} \rightarrow \mathcal{Y}$; (6) $\mathcal{Z}_k = \{\mathbf{z}_k^1, \dots, \mathbf{z}_k^{|\mathcal{U}_k|}\}$ are the electroencephalography (EEG) measurements/evidence given the true labels for the presented stimuli; (7) s is the stimulus (trial) index such that in epoch k , $|\mathcal{U}_k|$ number of stimuli are presented to the user before a decision is made; (8) the red box in this figure represents the conditional independence of the measurements in the set \mathcal{Z}_k . Note that in general the cardinality of the dictionary \mathcal{D} can be one and the cardinality of \mathcal{U}_k can be greater than the cardinality of \mathcal{D} . That is, same stimulus from \mathcal{D} can be presented to the user more than once in an epoch.

To be more specific, for example, in the typing testbed, in an epoch multiple sequences of symbols which are chosen from the dictionary \mathcal{D} are presented to a user, the BCI system attempts to make a decision after every sequence.

If a predefined confidence threshold is achieved a decision is made and the epoch ends, otherwise more sequences are shown until either the desired confidence or an upper bound on the number of sequences to be shown before a decision can be made is reached.

On the other hand, in the control testbed, dictionary \mathcal{D} consists of only one element which contains all the stimuli (flickering checkerboards). This translates to simultaneous presentation of the stimuli. A user is expected to choose one of these stimuli through the interface during each epoch. In this interface, an epoch consists of at least one trial. A trial is a complete presentation of the reversed-pattern checkerboards (stimuli) according to their control bit sequences. If a predefined confidence level is reached a decision is made, otherwise multiple trials are presented until either the confidence level or an upper bound on the number of trials before a decision can be made is reached. More specific explanations of these testbeds are provided in Sections 3 and 4.

Within this framework, we apply a maximum a posteriori (MAP) inference for fusion of the context information, the EEG evidence and joint intent detection such that

$$\hat{\mathbf{x}}_k^{MAP} = \arg \max_{\mathbf{x}_k} P(\mathbf{x}_k | \mathbf{c}_k, \mathcal{Z}_k; \mathcal{D}(\mathcal{U}_k)) = \arg \max_{\mathbf{x}_k} P(\mathbf{x}_k | \mathbf{c}_k) P(\mathcal{Z}_k | \mathcal{Y}_k(\mathbf{x}_k, \mathcal{D}(\mathcal{U}_k))) \mathbf{x}_k \quad (1)$$

The second equality is obtained by observing that conditioned on \mathbf{x}_k , \mathbf{c}_k and \mathcal{Z}_k are independent of each other. In Equation (1), $P(\mathbf{x}_k | \mathbf{c}_k)$ is the probabilistic information provided by the context information model, and $P(\mathcal{Z}_k | \mathcal{Y}_k(\mathbf{x}_k, \mathcal{D}(\mathcal{U}_k)))$ is the class conditional distribution of the electroencephalography (EEG) measurements/evidence. Using the assumptions presented in Figure 1, we can write

$$P(\mathcal{Z}_k | \mathcal{Y}_k(\mathbf{x}_k, \mathcal{D}(\mathcal{U}_k))) = \prod_{s=1}^{|\mathcal{U}_k|} p(\mathbf{z}_k^s | \mathbf{y}_k^s) \quad (2)$$

In Equation (2), marginal distributions of measurements \mathbf{Z}_k^s conditioned on the true labels \mathbf{Y}_k^s , $p(\mathbf{z}_k^s | \mathbf{y}_k^s)$, $p(\mathbf{z}_k^s | \mathbf{y}_k^s)$ can be parametrically or non-parametrically estimated from the labeled calibration data collected from the users and they can be input to the framework. Similarly, the posterior distribution of the state \mathbf{x}_k conditioned on the context information \mathbf{c}_k , $P(\mathbf{x}_k | \mathbf{c}_k)$ can be a probabilistic model that is externally estimated and input to the framework.

The generative model that is illustrated in Figure 1 is utilized for Monte Carlo simulations of different pre-designed tasks where the states of every epoch, true labels of every trial, and the set of stimuli indexed to be presented to the user, \mathcal{U}_k , are known. That is, for example, during epoch k , for every trial conditioned on its true value and state value, EEG evidence \mathbf{Z}_k^s is sampled from $p(\mathbf{z}_k^s | \mathbf{y}_k^s)$ for $s = 1, \dots, |\mathcal{U}_k|$. These sampled values are then used in

Equation (1) together with the estimated $P(\mathbf{x}_k | \mathbf{c}_k)$ for joint statistical state inference.

The developed simulation framework uses the assumptions imposed in Figure 1. These assumptions are generally made in the development of BCI systems. However, some of these assumptions, especially the independence of the EEG evidence corresponding to different trials in an epoch, may not be true. Moreover, during the simulation, the distributions $p(\mathbf{z}_k^s | \mathbf{y}_k^s)$ calculated using the calibration data prior to the testing of the systems are used throughout the simulation. This may not reflect the real interaction of the system with a user since the statistical characterization of the EEG may change during the operation. Over- or underestimation of system performance is possible. However, as we also show with our results, the proposed simulation framework provides a good approximation of the real-time system performance.

3. SSVEP-based control testbed

Visually evoked potentials (VEPs) are the responses to flashing stimuli occurring in the brain, specifically visual cortex. The peak of these responses occurs around 100 ms after the onset of the flash.[4] Different methods have been developed to take advantage of this phenomenon to create a communication or control interface. These methods, mainly, fall into the two categories *frequency-based* and *code-based*.[5] Frequency-based systems take advantage of the fact that visual cortex has a steady response called the steady-state visually evoked potential (SSVEP). These responses are stronger than normal VEPs and easier to detect. Hence, a constant frequency at which the stimulus flashes is usually used to induce such responses. On the other hand, code-based visually evoked potentials are the responses of the visual system to flashing stimuli, in which the flashes are not in a steady state and are controlled by a pseudorandom binary sequence. Depending on the length of the sequences used to control the flashes and the rate of the presentation, these stimuli can also create a strong and steady response which has a longer duration in time. In general, code-based systems tend to show a stronger, faster and more robust response in comparison with the frequency-based systems.

Here, we describe a multi-option code-based system implemented using the signal processing and simulation framework. Due to the high performance of this system we used only one EEG channel, Oz, placed right on the visual cortex.

3.1. Stimuli description

The stimuli consist of four reversed-pattern flickering checkerboards, placed at the four corners of a computer

display. While black and white remain a viable choice for the stimuli, a red and green color pair is used to generate even stronger responses.[6] The stimulation for each checkerboard is done by assigning zeros in the designated pseudorandom binary sequence to one pattern and ones to the reverse pattern. The checkerboards flash simultaneously, while controlled by different pseudorandom sequences. Here, we choose m-sequences of length 63 bits as the control sequences. The length of the sequences and the rate of the presentation are chosen based on previous studies.[7,8] The display refreshes at 110 Hz, leading to approximately half a second presentation time for presenting a complete sequence. During the stimulation, all the stimuli (checkerboards) are presented simultaneously. A complete presentation of the patterns associated with the bits in the control bit sequences is called a trial.

3.2. Operation modes

The system has three operation modes, *Calibration*, *Mastery Task* and *Free Run*.

Calibration: this is a session used to collect EEG evidence and build the templates corresponding to each stimulus being the target. To collect enough EEG evidence for each stimulus, the user is presented with 20 epochs. During each epoch, one checkerboard is selected as the target by placing a thin surrounding yellow frame. All the checkerboards are presented simultaneously. Each checkerboard is chosen as the target five times, while the order of the targets is randomly selected. During each epoch, 12 periods of the control sequences are presented. EEG signals are marked at the beginning of each period of the control sequences. While building the templates, EEG signals in response to the presentation of the first repetition of the control sequences are removed and considered as the visual cue time. Figure 2a shows a screen-shot of the Calibration session while the lower left stimulus is selected as the target.

Mastery Task: this is a session in which the user has to complete a predefined task. The main purpose of the Mastery Tasks is to train the user to use the system. Mastery Tasks can be generated in different difficulty levels. A novice user starts from the easy levels and can advance to a more difficult level upon successful completion of the task. The difficulty level is adjusted by modifying the context probabilities used as the prior probabilities fused with probabilities estimated using EEG evidence. Increasing the prior probability of the correct choice will increase the chance of the correct choice being classified as the intent, hence making the task easier. On the other hand, increasing the probability of the wrong choices will increase their chances of being classified as the intent and will make it harder for the user to choose the correct

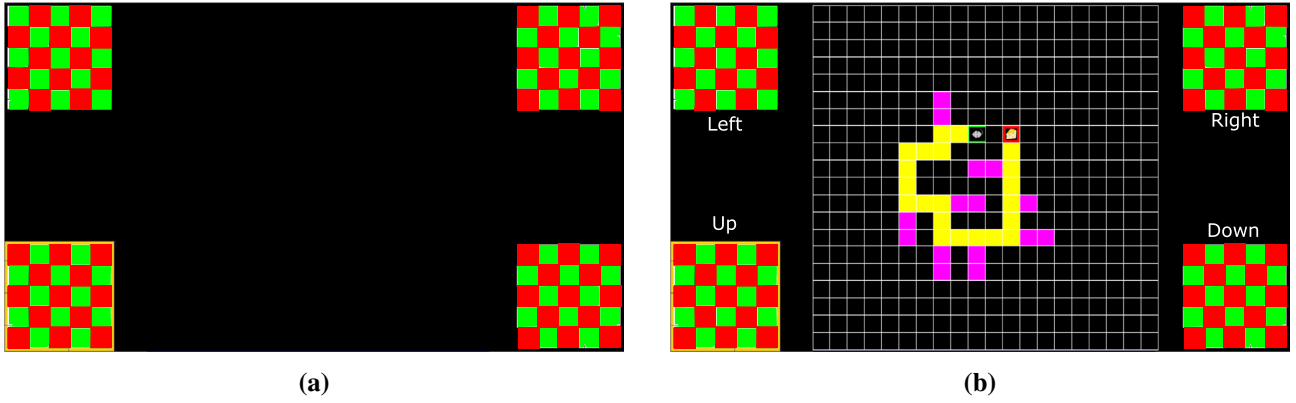


Figure 2. Screen-shots of the system during (a) Calibration and (b) Mastery Task.

choice. In this study, the difficulty levels are defined based on the ratio of the probability of the correct choice to the maximum probability of the wrong choices. Each mastery level applies a lower and an upper bound on the aforementioned probability ratios. In this study, going through a predefined path in a maze is used, as explained in Section 3.4. Figure 2b shows a sample maze as a Mastery Task placed at the center of the screen.

Free Run: this is the normal operation mode of the system in which the user has the freedom of choosing the next move as desired. Examples include controlling a wheelchair [9] and a robotic arm.

Both Mastery Task and Free Run modes require a calibration data set for each user. However, the calibration needs to be done only once and can be used for several sessions. For the purpose of this study, a calibration session was run once for every day of data collection.

During Mastery Task or Free Run modes, a perception of the environment is presented to the user in the center of the screen. Each time the user is given a chance to choose from the available options mapped to the four stimuli. Each stimulus on the screen has a designated text label, which presents its corresponding action. The users express their intent by focusing on the target stimuli. EEG data are processed in parallel, and the confidence levels of the decisions are compared with a static or dynamic threshold. Once the threshold is reached, the decision is sent to the target device and the user is notified by a yellow frame surrounding the corresponding stimulus. The stimulation pauses until the target device finishes executing the task. When the target device is ready for the next command, the stimulation starts and the same procedure continues.

3.3. Signal processing and feature extraction

EEG signal processing starts by bandpass filtering and extracting the time-locked responses to the target stimuli. After the calibration session, EEG data are filtered

using a bandpass FIR filter from 2.5 Hz to 100 Hz and separated into four classes based on the target stimulus of each epoch and the onsets of the control sequences. This procedure results in 50 good-quality trials for each stimulus. Note that a trial is a complete presentation of the reverse pattern checkerboards associated with all the bits in the control bit sequences.

3.3.1. EEG feature extraction

A template response for each checkerboard is estimated by taking the median from the time-locked samples of EEG. Using the median, instead of simply taking the average, helps to eliminate the outliers in the noisy EEG data.

Next, by applying k -fold cross-validation on the calibration data, template matching scores are generated for each trial. During this phase, templates are built using $(k-1)$ folds and applied on the remaining fold. Template matching scores for each channel form an N_s dimensional vector, where N_s is the number of stimuli and every element in the vector is obtained by calculating the cross-correlation between the target trial and the template built for each stimulus.

The correlation score, r_i^c , between template for i th stimulus for channel c , t_i^c , and the windowed EEG signal from that channel, e^c , is given by

$$r_i^c = (e^c)^T t_i^c \quad (3)$$

$\mathbf{r}^c = [r_1^c, r_2^c, \dots, r_{N_s}^c]$ is the corresponding correlation vector for channel c . [7] Here, we used a single channel, Oz, hence $c = 1$.

Next, an $N_s \times N_c$ dimensional density is built for each stimulus based on the score vectors obtained in the previous step, where N_c is the number of EEG channels used in the classification. Including the correlation scores of the windowed EEG with the target stimulus template along with the cross-correlation scores between the windowed EEG and the non-target stimuli templates models the

variation of the scores simultaneously making the likelihoods more informative.

$$p(\mathbf{r}) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left[-\frac{1}{2} (\mathbf{r} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{r} - \boldsymbol{\mu}) \right] \quad (4)$$

where $\mathbf{r} = [\mathbf{r}^1, \dots, \mathbf{r}^{N_c}]$, $\boldsymbol{\mu} = [\mu_1, \dots, \mu_{N_c}]$ mean vector and Σ is the $N_s N_c \times N_s N_c$ covariance matrix with $|\Sigma|$ and Σ^{-1} being its determinant and inverse.[10] Densities can be built using different methods, such as kernel density estimation (KDE) or estimating the density parameters considering multi-dimensional densities. Here we have considered multi-variate Gaussian (MVG) densities with maximum likelihood estimates for mean and covariance. This is an important step. A better estimate of the actual densities will result in better estimation of the intended stimulus and also more accurate simulation results. These densities are used to calculate the class conditional probabilities in Equation (2).

3.3.2. Classification and decision-making

Classification is done using the maximum a posteriori classifier described in Equation (1). Next we describe how the graphical model presented in Figure 1 applies to this testbed. In this testbed, the dictionary \mathcal{D} has four elements representing all the checkerboard stimuli. In epoch k , \mathcal{U}_k includes one element which contains all the members of \mathcal{D} multiple times. This comes from the fact that the stimulus is the same for all the trials. \mathbf{x}_k is the user-intended stimulus; and \mathbf{Z}_k^i is the EEG evidence collected in response to the i th trial (r_i^c). Note that in this testbed \mathbf{y}_k^i and \mathbf{x}_k are the same.

Context information, $P(\mathbf{x}_k | \mathbf{c}_k)$, presented in Equation (2), depends on the application. For example, in a wheelchair control application, it might come from the map of the environment; similarly, for the Maze game described in Section 3.4, it is calculated based on the feasible directions at each point. In a Mastery Task, these probabilities are adjusted based on the difficulty level.

The following assumptions make it possible to use the MAP inference in Equation (1). First, it is considered that the target stays the same during each epoch. Second, it is considered that each epoch has only one target. Third, trials in an epoch are considered independent of each other.

During the Mastery Task or Free Run modes, template matching scores are evaluated on the class conditional densities for each stimulus and the likelihood scores obtained are fused with the context information, if available, based on Equation (1).

3.4. Task description

While the same system has been successfully used to control an actual wheelchair, a remote rover and a robotic

arm, in this study we chose a virtual task. In this virtual task, four options are presented to the user at each time representing four different directions, *Left*, *Right*, *Up* and *Down*. The users are presented with a virtual maze placed at the center of the screen with no overlap with the stimuli, as shown in Figure 2b.

Each maze has only one correct path connecting the *Start* point highlighted in green to the *End* point highlighted in red with several dead-end incorrect paths along the way. The correct path is highlighted in yellow and the incorrect paths are highlighted in pink. Figure 2b shows a sample maze placed at the center of the screen.

Users are asked to control the movements of a mouse icon from the Start point to an icon showing a slice of cheese place at the End point. If they go through an incorrect path, the users need to correct their mistakes by moving the mouse icon back to the correct path. The number of valid options at each point on the maze depends on the number of paths connecting to that point. Based on the design, at each point on the maze the number of valid choices can be one, two, three or four.

The virtual mazes are used to build the context probabilities based on the valid choices at each point. Directions facing a wall are given very low probabilities and directions towards feasible paths are given probabilities according to the task settings.

3.5. Performance estimation via simulations

Ten healthy individuals, six females and four males, with normal or corrected vision between the ages of 23 and 30 consented and participated. Data collections were performed based on an approved IRB protocol at Northeastern University. Participants were seated comfortably in front of a laptop screen around 65 cm from the screen. EEG was recorded using a g.USBamp biosignal amplifier using active g.Butterfly electrodes from g.tec (Graz, Austria) at a 256 Hz sampling rate. A single EEG electrode was applied using the g.GAMMAcap (electrode cap) and positioned according to the International 10/20 system with location Oz. This electrode was grounded with respect to FPz and referenced against an earclip placed on the right ear lobe.

Each participant attended one data collection session. This session started with a Calibration and followed with four Mastery Tasks. Figure 3 illustrates four mazes presented during the Mastery Tasks in this experiment. In the Mastery Tasks, the prior probability distribution over the control options was assumed to be uniform. Under these assumptions, four mazes were randomly generated and each participant completed the Mastery Task over these four mazes. Accuracy and number of trials per target are computed as the performance measures using the real-time experimental results. Actual performance, actual

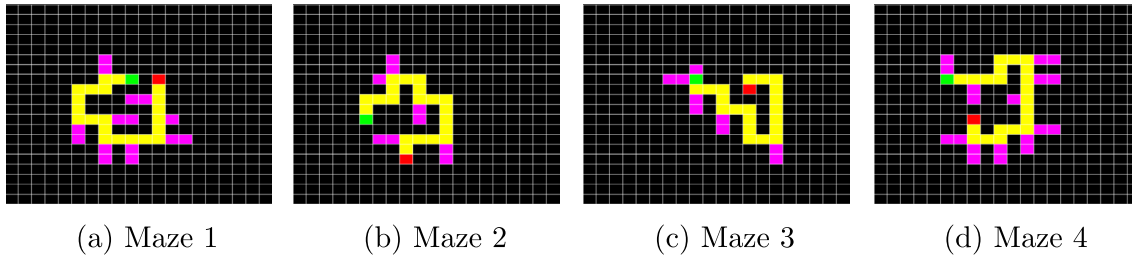


Figure 3. Mazes used during the four Mastery Tasks.

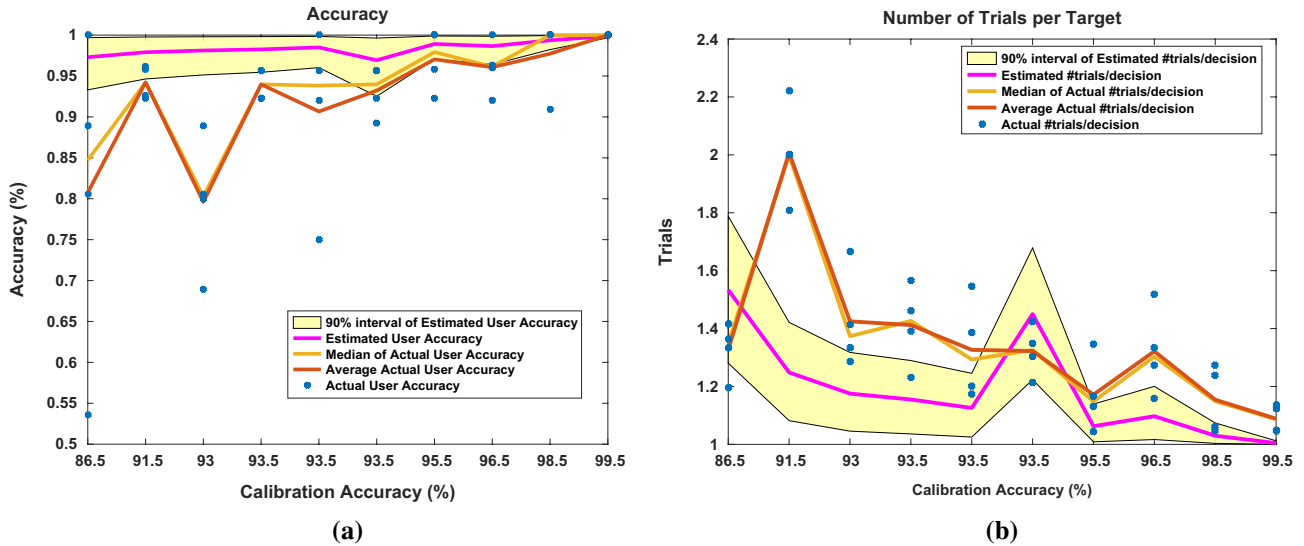


Figure 4. Estimated and actual performance measures: (a) accuracy and (b) number of trials per target.

average and median performance results are presented in Figure 4 as a function of calibration accuracy (area under the receiver operating characteristic (ROC) curve (AUC)).

In addition to the real-time experimental results, using the calibration data for each participant, the performance measures were estimated through the simulation framework. For each participant, 10,000 Monte Carlo simulations were performed. The estimated performance and 90% estimation confidence intervals are presented in Figure 4 as a function of calibration accuracy.

From Figure 4, comparing the actual performance results (blue dots) with the estimated ones, we observe that under and over performance estimations are possible. We conducted a two-tailed Wilcoxon signed-rank test between simulation and experimental results. Since the median is more robust to outliers, we compared the median of the actual performance with the estimated performance values. The test results showed that with a p -value of .01, the simulation overestimated the actual accuracies, but with a p -value of .43 the actual accuracies are not significantly different from the lower 90% confidence level of the estimated values. Moreover, with a p -value of .11, the estimated numbers of trials per target are not significantly different from the actual values.

4. RSVP Keyboard testbed

The RSVP Keyboard is a novel language-model-assisted electroencephalography (EEG) based brain-computer interface (BCI) for letter-by-letter typing.[11,12] The RSVP Keyboard is based on a rapid serial visual presentation paradigm that is not strongly dependent on gaze control RSVP is a visual stimulation methodology relying on the temporal separation and rapid succession of visual items at a fixed focal area on a screen.[11–16]

The RSVP Keyboard currently enables the selection from an alphabet of 28 symbols: 26 letters in the English alphabet, and two symbols for space and backspace. It is possible to extend the set of symbols to contain additional symbols for word completion, punctuation or various commands. Presenting icons representing words could also be included and we have a working prototype with this variation with appropriate modification of the language model; this variation is outside the scope of this paper. A *trial* is the presentation of one symbol, which represents a single stimulus. A *sequence* is a burst of trials that present several symbols (the entire alphabet or a carefully selected subset) in RSVP mode. During the presentation of a sequence, currently restricted to not present any symbol

more than once, the symbols are ordered randomly for the purpose of strengthening the response for the intended symbol; however, recent evidence from auditory P300 BCIs indicates that randomization may not be crucial for successful operation.[17] The selection of a single symbol in the typing process occurs in an *epoch*, a period that consists of one or more sequences depending on when sufficient confidence is built based on the language evidence and accumulated multiple EEG evidence for candidate symbols. Simulations based on calibration EEG data are used to optimize parameters (such as max-sequence-per-epoch) to improve speed and accuracy during use. During the epoch, the user is expected (assumed) to react positively for and only for the target symbol. As such, the inference problem reduces to deciding between positive and negative intent for each symbol in the alphabet.

As a response to the infrequent, salient stimulus in RSVP, an event-related potential (ERP) is generated in the brain. Prominently it shows in the EEG as a positive deflection in the scalp voltage potentials with a rough latency of 300 ms, named as P300, predominantly in the centro-parietal regions. However, the features of the P300 waveform, e.g. regions, latency, amplitude and shape, may differ for each user. This natural target-stimulus-recognition response of the brain allows us to design BCI systems by sensing them.

4.1. EEG feature extraction

EEG evidence needs to be extracted from the aforementioned P300-based novelty detection or target-recognition ERP in order to allow the detection of user intent from EEG signals. This process starts with extracting stimulus-time-locked bandpass (BP) filtered EEG signals for each stimulus in the sequence. Since, physiologically, the most relevant non-motor signal components are expected to occur within the first 500 ms following the stimuli, the [0, 500) ms portion of the EEG following each stimulus is extracted. The restriction of the time window by 500 ms also enables omitting most of the related unintended motor cortex responses, if the healthy participants engage in intentional or unintentional motor activity following the target stimuli. For locked-in users with motor capabilities, this restriction could be relaxed to allow the system to benefit from any such motor responses for an improved performance (at the cost of not truly being a brain-computer interface any more).

We employ the following procedure to extract features that are well separated. (1) Temporal-windowed EEG signals are filtered by [1.5,42] Hz bandpass filter (FIR, linear phase, length 153, 0 DC-gain) to eliminate the low-frequency deviations and high-frequency noise. Lower high-cut-off frequencies may be used. (2) For each channel individually, signals corresponding to the [0, 500) ms post-stimulus

interval (after taking into account the group delay of the BP filter) are vectorized and projected to a slightly lower-dimensional space by linear dimension reduction. In the existing system, this is achieved by principal-component analysis (PCA) to remove directions with negligible variance (as determined by calibration EEG data). The principal directions with variance lower than 10^{-5} times that of the first principal direction are removed. (3) Projected vectors corresponding to each channel are concatenated to generate a single aggregated feature set. This process constitutes a linear orthogonal projection of raw spatio-temporal samples eigenfiltering and downsampling. (4) Using regularized discriminant analysis (RDA),[18] a quadratic projection of this aggregate feature vector to K_h is achieved. This reduces the dimensionality of EEG evidence for each trial to one in the current implementation.

There exist numerous ways to preprocess and classify the ERP responses.[19,20] It is possible to design another feature-extraction methodology to replace the existing one with an improved performance. The primary objective of this paper is not to find the best feature-extraction procedure, but to exhibit the potency of the overall design choices including tight probabilistic fusion with context information. The entire raw-EEG-to-feature pipeline could be improved, and this is the subject of future research and development.

4.1.1. Regularized discriminant analysis (RDA)

RDA is a model-based analysis technique, which is an extension of quadratic discriminant analysis (QDA) to counteract the curse of dimensionality.[18] If the class feature vectors are assumed to conditionally belong to multivariate normal distribution, the log-likelihood ratio becomes a quadratic function. Correspondingly, the optimal Bayes decision rule belongs to the QDA family. Consequently, under the Gaussianity assumption, QDA projection in step (4) above yields the best one-dimensional feature in the minimum expected risk sense. It only depends on the class means, covariance matrices, class priors, and a relative-risk-based threshold/weighting (threshold if only EEG evidence is used to classify). The distribution parameters are estimated from the training data, in this case from a calibration session which is a supervised data collection session to learn the signal statistics. When the number of samples is smaller than the number of dimensions, the maximum likelihood estimates of the class covariance matrices become rank deficient. Due to the time constraints on the calibration sessions, this situation typically exists for the analysis of ERPs. Since the inverse and determinant of the covariance estimates are needed for the corresponding QDA solution, singularities on those prevent the applicability of the analysis.

RDA is designed as one method to countermeasure the curse of dimensionality. Shrinkage and regularization procedures are applied on the covariance estimates to reduce the rank deficiency.[10] Let $\hat{\boldsymbol{\mu}}_k$ and $\hat{\boldsymbol{\Sigma}}_k$ be the maximum likelihood estimates for class means and covariances, respectively, where $k \in \{0,1\}$ corresponds the class label. Correspondingly, RDA updates the estimates of the class covariances to be $\hat{\boldsymbol{\Sigma}}_k(\lambda, \gamma)$ with λ and $\gamma \in [0,1]$ as the shrinkage and regularization parameters, respectively.[18] Then accordingly, the RDA score for feature vector ω is computed as

$$z(\omega) = \log \frac{f_N(\omega; \hat{\boldsymbol{\mu}}_1, \hat{\boldsymbol{\Sigma}}_1(\lambda, \gamma)) \hat{\pi}_1}{f_N(\omega; \hat{\boldsymbol{\mu}}_0, \hat{\boldsymbol{\Sigma}}_1(\lambda, \gamma)) \hat{\pi}_0} \quad (5)$$

where z is the EEG evidence as described in Figure 1, $\hat{\pi}_k$ is the estimate of class priors, and $f_N(\omega; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the pdf of a multivariate normal distribution. In the RSVP Keyboard, the result of the nonlinear projection via RDA, i.e. z , is used as the one-dimensional EEG feature for fusion with language models as explained in Section 4.3.

4.2 Estimation of the score conditional probability density functions

Finally, the conditional pdf of z given the class labels, i.e. $f(z|y = k)$, needs to be estimated for utilizing them in a probabilistic framework. Estimation is achieved nonparametrically using kernel density estimation (KDE) on the training data as

$$\hat{f}(z|y = k) = \frac{1}{N_k} \sum_{y(v)=k} K_{h_k}(z - z(v)) \quad (6)$$

where $K_{h_k}(\cdot)$ is the kernel function with bandwidth h_k , and $z(v)$ is the discriminant score, obtained by Equation (5), corresponding to a sample $v \in \{1, 2, \dots, N\}$ in the training data. $z(v)$ is estimated via cross validation. For KDE estimation we use the Gaussian kernel and estimate the bandwidth h_k using Silverman's rule of thumb [21] for each class k . Silverman's rule minimizes the mean square integrated error while assuming that the underlying density has the same curvature as its variance-matching normal distribution.

The conditional density functions estimated here are used in Section 4.3 for the probabilistic context incorporated decision-making and in Section 4.5 for estimating the typing performance via simulations.

4.3. Joint decision-making

The evidence obtained from EEG and the language model is used collaboratively to make a more informative

symbol decision. An epoch, the combination of sequences that consist of trials corresponding to different symbols, is shown for each symbol to be selected. The collected EEG evidence from an epoch is fused with the language model to make the final decision on the symbol selected by the BCI user. The fusion model is described below.

4.3.1. Fusion rule

Let $y : \Omega \rightarrow \{0,1\}$ be the random variable representing the class of intent, where 0 and 1 corresponds to negative and positive intents, respectively, and $\mathbf{z} : \Omega \rightarrow \mathbb{R}^d$ be a random vector of EEG features corresponding to a trial. For example, an ERP discriminant function that projects the EEG data corresponding to a trial into a single dimension may be used as a feature-extraction method. In this case, $d = 1$, which denotes that there is only one EEG feature per trial. The fusion methodology explained here does not depend on the feature-extraction method, and practically can be used with any feature vector in \mathbb{R}^d . The only requirement is an estimate of the conditional probability density function of EEG features given the class label, i.e. $p(\mathbf{z}|y = c) \forall c \in \{0,1\}$.

Specifically, let $z_{t,x,r}$ be the random EEG evidence vector corresponding to a trial for epoch $t \in \mathbb{N}$, symbol $x \in X$ and repetition $r \in \{1, 2, \dots, R_x\}$, where R_x is the total number of repetitions of symbol x in epoch t . Furthermore, let $y_{t,x}$ be the random variable representing the class of epoch t and symbol x . under the assumptions of the graphical model presented in Figure 1

$$P(x_t = x | \mathcal{Z}_t, c_t; \mathcal{D}(\mathcal{U}_t)) \propto p(\mathcal{Z}_t | \mathcal{Y}_t(x_t, \mathcal{D}(\mathcal{U}_t))) P(x_t | c_t), \quad (7)$$

where $\mathcal{Z}_t = \{z_{t,x,r}\}$ for all $x \in X$ and $r = 1, \dots, R_x$ is the set of all EEG evidence obtained for the t th epoch, \mathcal{Y}_t is the set of labels corresponding to the trials in epoch t , \mathcal{D} is the set of all possible symbols and \mathcal{U}_t is a set of indexes that identify the symbols presented in epoch t . Moreover $P(x_t | c_t)$ represents the language model probability. The language model is an n -gram model trained from a one-million-sentence sample of the *NY Times* portion of the English Gigaword corpus.[11,22,23] Importantly, the corpus was case-normalized and, we used Witten-Bell smoothing for regularization.[24] The details of the corpus normalization and smoothing methodology are described in [25].

4.3.2. Final decision

After applying one of the fusion rules the intended symbol is decided by selecting the symbol with maximum probability given the EEG and context evidences,

$$\hat{s} = \underset{x}{\operatorname{argmax}} P(x_t = x | \mathcal{Z}_t, c_t).$$

At the end of each sequence $P(x_t = x | \mathcal{Z}_t, c_t)$ is used as a confidence measure. Using the maximum value in the probability mass function as the measure of certainty corresponds to utilizing Renyi's order-infinity entropy definition as the measure of uncertainty in the decision we make. One could use other definitions of entropy to measure uncertainty/certainty in the current decision. Decision in the epoch is finalized if one of the following conditions is satisfied:

- $\max_x P(x_t = x | \mathcal{Z}_t, c_t) > \gamma$, where γ is the confidence threshold that is currently set to 0.9.
- the number of sequences reaches the maximum number of sequences allowed per epoch, e.g. 4, 8, 16.

4.3.3. Deciding on the next sequence

After each sequence the posterior probabilities for the symbols are calculated. If the number of trials per sequence is smaller than $|S|$, the symbols to be shown in the following sequence are decided either randomly or by selecting the most likely symbols according to the current posterior probabilities. In the mode that selects the most likely symbols, it might potentially prevent context-unlikely symbols being shown. However, they are expected to show up as the subject demonstrates negative intent for the rest. Indeed, the subjects were even able to select very context-unlikely symbols such as Q.

4.4. Operation modes

The RSVP Keyboard has the following operation modes.

Calibration: the session in which the signal statistics for the participant are learned. A predefined target symbol is selected and shown to the user before an RSVP sequence. Participants are instructed to look for the target symbol in the sequence and show intent for the predefined symbol. Using the collected supervised data, the EEG feature-extraction parameters are optimized. Typing performance might be estimated using the approach proposed in Section 4.5.

Free Spelling: free spelling is the main typing mode. The participants can type whatever they want and use the system to communicate.

Mastery Task: a Mastery Task is designed to train the users on using the system. Users are asked to complete missing words in different phrases. The phrases are divided into levels with increasing difficulty. The difficulty of the phrases is assessed by the ratio of the context probability of the intended symbol to the unintended most likely symbol for each letter in the phrase. Each level consists of multiple sets of phrases. Subjects pass through levels by successfully completing two-thirds of the phrases in a set.[26]

4.5. Performance estimation via simulations

Not all parameters can be easily theoretically optimized in a complex system. Experimentation on human subjects is a very time-consuming process, therefore trial and error with the user in the loop is not feasible. In order to be able to get a reliable performance estimate for a setting, one might need to do multiple sessions that might take more than an hour each. If the researcher/clinician would like to do comparative experiments on different operation scenarios, e.g. number of maximum sequences per epoch, backspace probability, bandpass cut-off frequencies, etc., than the experiments might take exponentially increasing amounts of time (by parameter space dimensionality, for grid search). The number of sessions needed might exponentially grow with the number of parameters being searched.

As a supplement to experimentation, we utilize a simulation methodology to estimate the performance of the system. In simulation, the copy phrase task is employed, since the user would need to attempt perfect phrase-copying and the user strategy can be simulated exactly; see Section 4.5.1. In other words, the simulated user in the simulation always attempts to correct any errors. Consequently, at any instant the intended action is known. Afterwards, RDA projected one-dimensional EEG features are sampled from Equation (6), the conditional pdf estimates learned after cross validation. The features corresponding to the trials of the intended symbol are sampled from the target pdf and the rest are sampled from the non-target pdf.

The simulations are based on the assumption of i.i.d. target and non-target trial EEG evidence. They ignore potentially different responses to different non-target symbols (nonstationarity). They ignore fatigue effects or change in signal statistics or temporal dynamics of the EEG due to other physiological reasons, as well. However it is still a very beneficial tool for estimating the performance. In our experience, actual copy phrase results with subject-in-the-loop are usually consistent with Monte Carlo simulation estimates.

4.5.1. Copy phrase

Copy phrase mode is a guided typing mode. It can be used either in simulation or in real-time typing. In this mode, participants are asked to copy a phrase in a sentence letter by letter. Participants are also instructed to delete any of their mistakes. Copying continues to the next phrase if one of the following conditions is satisfied:

- Exactly copying the phrase (success).
- Timeout (failure): reaching a predefined time allowed for the phrase.

- Sequence Limit (failure): the number of sequences spent is limited by $\kappa \times$ (number of letters in phrase) \times (maximum number of sequences per epoch), where κ is a scale parameter, which is currently set to 2.
- Consecutive erroneously typed symbols (failure): if there exist three consecutive unintended letters in the typed text, it is considered as failure.

To assess the performance of the participant in the copy phrase task, the following metrics are utilized:

- Probability of Successful Completion: the ratio of the phrases which are correctly completed within the allocated sequence and time limits. Any uncorrected error would cause the phrase to be considered unsuccessfully completed.
- Total Duration of the Copy Tasks: the total time spent to go through all the phrases, successfully or unsuccessfully.

4.6. Experiments

Ten healthy participants (four female and six male, aged 24–37) were recruited in accordance with Northeastern IRB 13-01-07. Correspondingly each participant gave informed consent and continued in the experiments for five or six sessions. The first session was a calibration session with 100 randomly selected sequences of 10 trials with 200 ms inter-stimulus intervals. If the single trial area under the ROC is lower than 0.7, the calibration session is repeated once again. Subsequently, four copy task sessions with following parameter selections are conducted to collect experimental data.

- Scenario 1: no language model, 28 trials/sequence, at most 4 sequences/epoch
- Scenario 2: 6-gram model, 28 trials/sequence, at most 4 sequences/epoch
- Scenario 3: 6-gram model, 14 trials/sequence, at most 8 sequences/epoch
- Scenario 4: 6-gram model, 7 trials/sequence, at most 16 sequences/epoch

In scenarios 3 and 4, the set of symbols in each sequence is selected according to the posterior probabilities as in Section 4.3.3. In addition to the experimental results, using the calibration data collected from each participant, we applied the simulation module to all four of these scenarios.

In the copy task the following eight phrase and sentence pairs are presented to be copied. These phrases and sentences were randomly selected from the Mackenzie and Soukoreff phrase sets for evaluating text entry techniques.[27]

- THE DOG **WILL** BITE YOU
- LEARN TO WALK BEFORE YOU **RUN**
- GOOD AT ADDITION AND SUBTRACTION
- INTERACTIONS **BETWEEN** MEN AND WOMEN
- ACCOMPANIED BY AN **ADULT**
- **PLEASE** KEEP THIS CONFIDENTIAL
- FISH **ARE** JUMPING
- SHE WEARS TOO MUCH **MAKEUP**

These sentences and target phrases are randomly selected from MacKenzie and Sourkeff's balanced phrase list. This list was prepared by MacKenzie and Sourkeff and contains 500 phrases that are moderate in length, easy to remember and representative of the target language.

Participants were seated comfortably in front of a laptop screen, around 65 cm from the screen on which each presented stimulus spanned 4.5° of visual angle. EEG was recorded using a g.USBamp biosignal amplifier using active g.Butterfly electrodes from g.tec (Graz, Austria) at a 256 Hz sampling rate. The EEG electrodes were applied using the g.GAMMAcap (electrode cap) and positioned according to the International 10/20 system with locations Fp1, Fp2, F3, F4, FZ, Fc1, Fc2, Cz, P1, P2, C1 C2, Cp3, Cp4, P5, P6. They were grounded with respect to FPz and referenced against an earclip placed on the right earlobe. During the setup process, impedances were kept below 7 k Ω using the Simulink impedance measurement tool from g.tec.

Internal to the amplifier, signals were filtered by a nonlinear-phase [2, 60] Hz Butterworth bandpass filter and a 60 Hz notch filter. Afterwards, signals were filtered further by the previously mentioned [1.5, 42] Hz linear-phase bandpass filter (our design). The filtered signals were downsampled to 128 Hz. For each channel, stimulus-onset-locked time windows of [0, 500) ms following each image onset were taken as the stimulus response and features were extracted as described earlier.

The grand average of ERPs for the P1 and P5 channels is given in Figure 5. Since the stimulus rate is 200 ms per symbol, a strong visual N100 response is observed for both the current stimulus and the following one in the ERP averages of the non-targets. N200 and P300 components constitute the components distinguishing between target and non-target responses.

Results of the experiments (one user-in-the-loop session per scenario) and simulations (mode of 50 Monte Carlo runs) are given in In Figures 6 and 7. Comparing the results of Scenario 1 to the other scenarios, we observe that the language model improves both the accuracy and the speed of typing. Moreover, comparing the results of Scenarios 2, 3 and 4, we illustrate that decreasing the number of symbols in a sequence and increasing the maximum number of sequences in an epoch in general increases

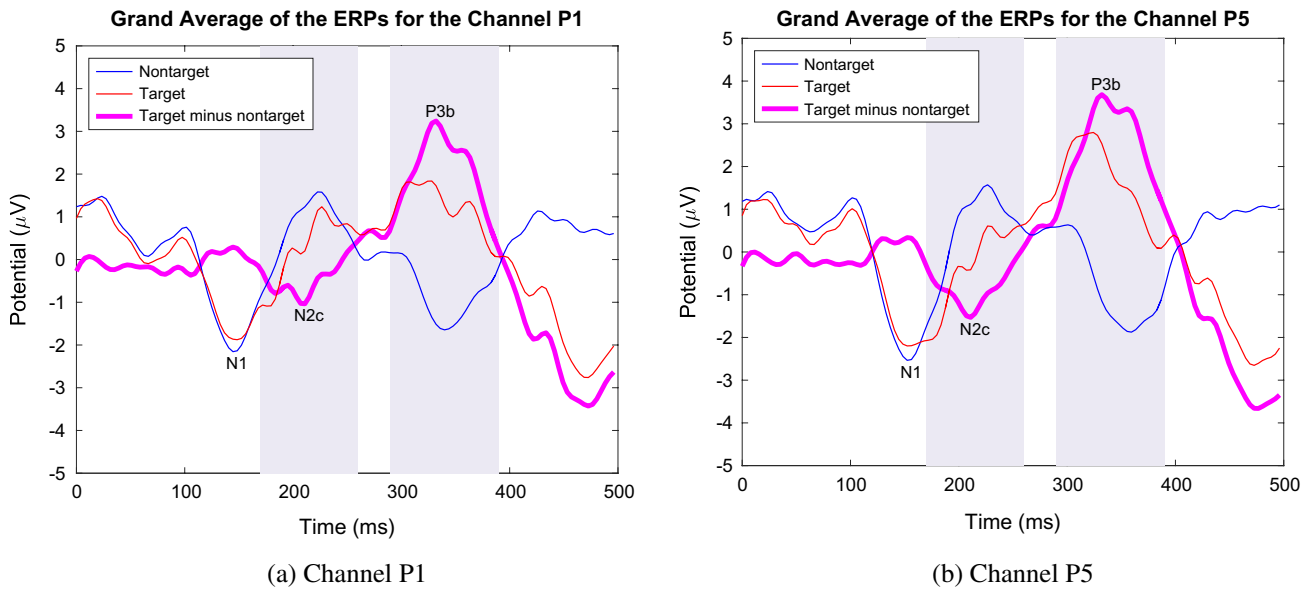


Figure 5. Grand averages of ERPs for the P1 and P5 electrodes.

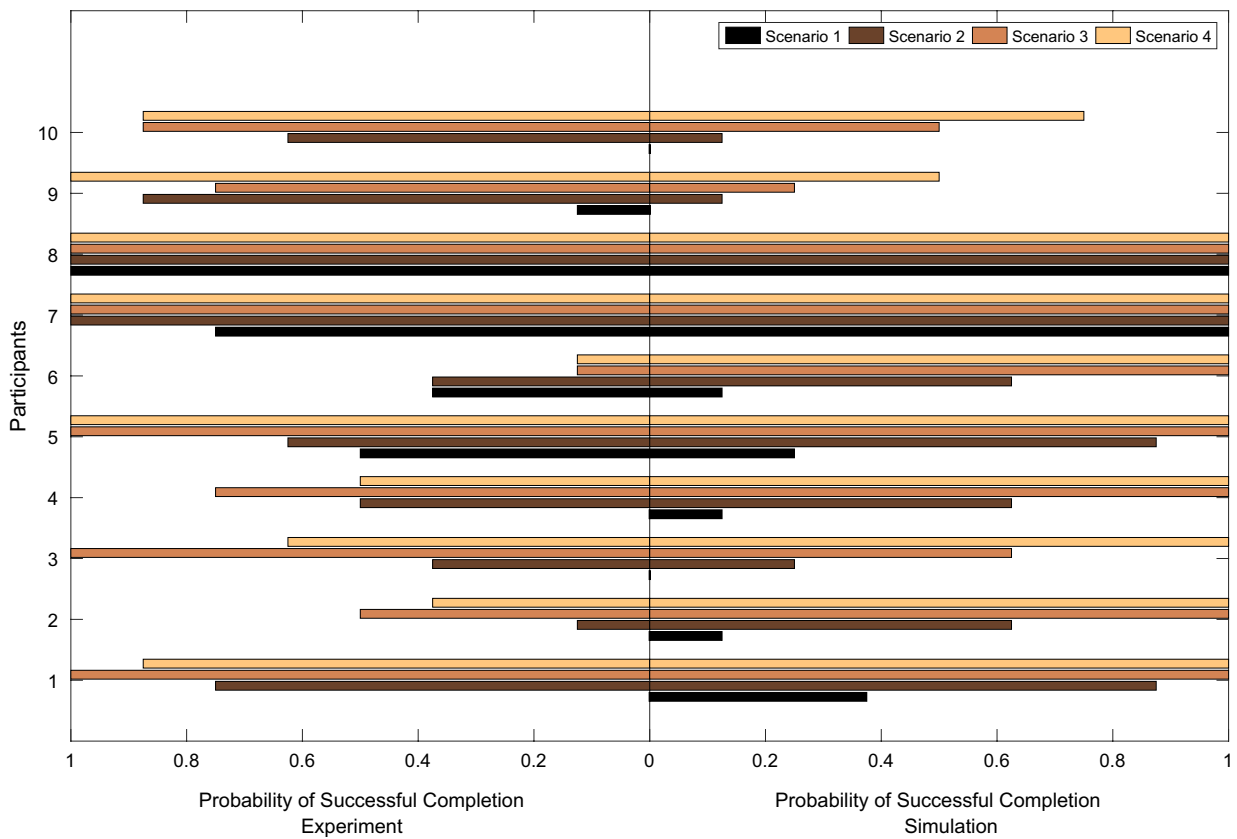


Figure 6. Successful completion probability for each subject for each scenario for simulation and experiment. Right and left panels correspond to simulation and experimental results, respectively.

the typing performance. To quantitatively assess these improvements we applied a one-sided Wilcoxon signed-rank test, which is a nonparametric test analogous to

the paired Student's *t*-test, between pairs of scenarios. Correspondingly, statistical results also show that the inclusion of the language model introduces a significant

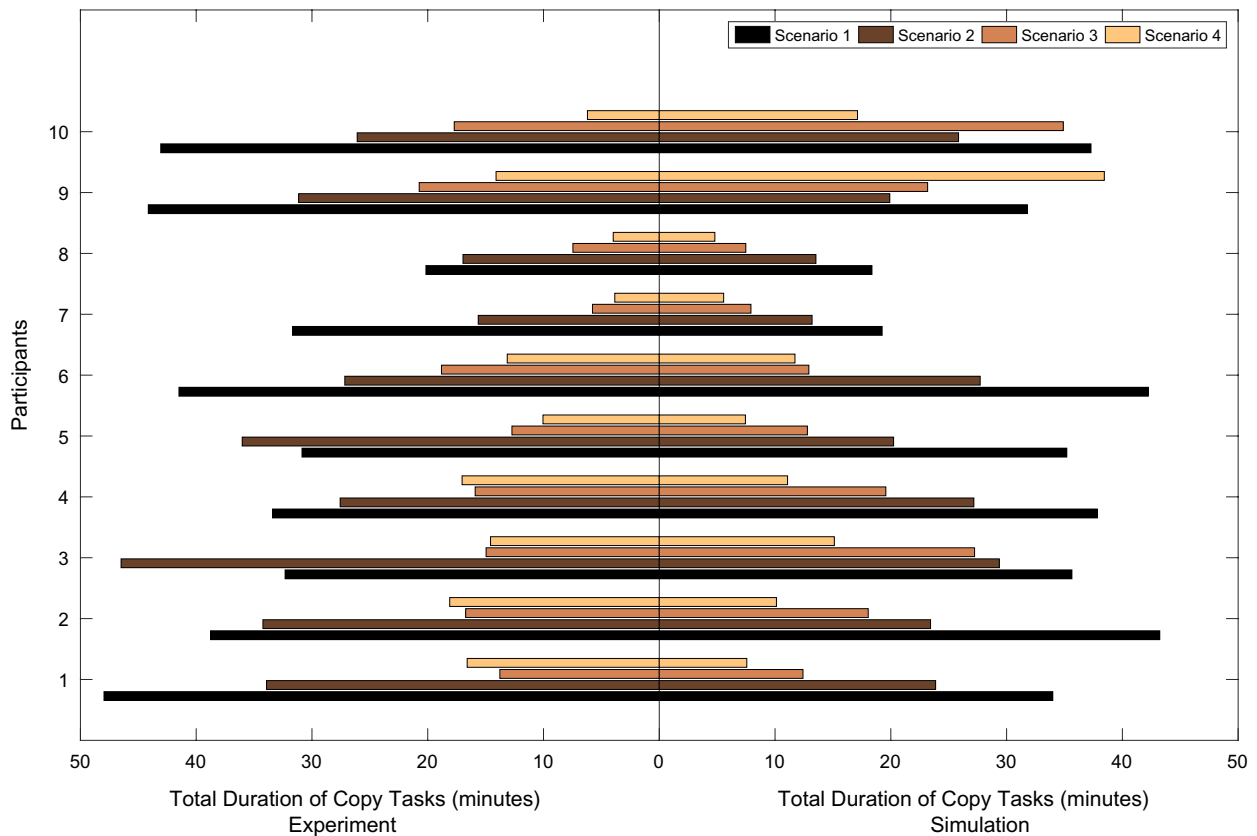


Figure 7. Duration to copy eight phrases for each subject for each scenario for simulation and experiment. This corresponds to the total duration spent on attempting to type all of the phrases. Right and left panels correspond to simulation and experimental results, respectively.

Table 1. Summary of the statistical test results: This table contains the *p*-values comparing various scenarios in both experiments and simulations using a one-tailed Wilcoxon signed-rank test. Duration and accuracy correspond to total duration of copy tasks and probability of successful completion, respectively. In all statistical tests, the alternative hypothesis corresponds to one scenario being larger than another one in either duration or accuracy and the significance level was selected to be 5%.

Accuracy	H_1	S1 < S2	S1 < S3	S1 < S4	S2 < S3	S2 < S4	S3 < S4
Exp <i>p</i> -values	Sim	.004*	.006*	.006*	.039*	.063	.875
		.004*	.004*	.004*	.004*	.004*	.125
Duration	H_1	S1 > S2	S1 > S3	S1 > S4	S2 > S3	S2 > S4	S3 > S4
Exp <i>p</i> -values	Sim	.042*	.001*	.001*	.001*	.001*	.053
		.001*	.001*	.002*	.042*	.042*	.032*

difference in the probability of successful completion and total typing duration for both experiments and simulations. The differences between Scenario 1 (no language model) vs. Scenario 2, Scenario 3 and Scenario 4 are all significant. Decreasing the number of trials per sequence from 28 to 14 also constitutes a significant difference. However, decreasing it further to seven trials per sequence did not demonstrate a statistically significant improvement in the performance. A summary of these results is given in Table 1.

Comparing the simulation and experimental results, as expected, we observe both under- and over-representation of the performance due to possible non-stationarities in the recorded EEG during the experiments. However, the

simulation tool provides in general a good estimate of typing performance. To support this hypothesis we conducted a two-tailed Wilcoxon signed-rank test between simulation and experimental results. The tests did not find any statistically significant difference in the total durations and probability of successful completion, with large *p*-values of .265 and .325. This demonstrates that the simulation is a useful tool which successfully tracks the changes in the experimental performance. Correspondingly, one of these scenarios, i.e. a set of heuristic parameter settings, might be selected through simulations to be used in the typing operation.

The mean and the best results of the RSVP Keyboard are presented in Table 2 and these results are compared

Table 2. Comparison with other BCI typing systems in the literature.

		Best	Best	Mean	Mean
		(bpm)	(sym/min)	(bpm)	(sym/min)
RSVP Keyboard	RSVP	31	6.4	10	2.2
Townsend et al. [28]	Matrix	39	6.3	31	5.1
Serby et al. [29]	Matrix	–	6.9	–	4.5
Trader et al. [30]	Matrix	–	–	9.8	2
Blankertz et al. [31]	Hex-o-spell	–	4.6–7.6	–	–
Acqualagna et al. [13]	RSVP	7	1.4	–	1.4
Speier et al. [32]	Matrix	53	11	37	9
Ryan et al. [33]	Matrix	30	5	19	3.7
Mainsah et al. [34]	Matrix	48	–	32	–
Furdea et al. [35, 36]	Auditory	12	2.7	7	1.28
An et al. [37]	Auditory and visual	–	–	10	1.65

with the results of the other leading efforts in online typing BCIs in the literature. The comparisons are based on the true bit-rates. Note that while the comparisons are made, the differences in the stimulation methodology (e.g. tactile, auditory or visual) and experimental conditions need to be considered. For example, these systems may experimentally differ due to gaze dependency, subject inclusion criteria or text used for estimating the performance. Additionally, some of the systems do not require error-correcting during the typing process, but try to estimate the additional load corresponding to erasure. Keeping these in mind and according to the results summarized in Table 2, the maximum typing speed of the RSVP Keyboard was at least on par with other gaze-independent BCI typing systems. However the performance of the RSVP Keyboard is worse than the matrix-based spellers.

5. Discussion

In this paper, we have introduced a simulation module to estimate the accuracy and speed of BCI systems. We discussed how this framework could be utilized for BCI system design to make small design choices rather than following time-consuming and expensive experimental procedures. In order to demonstrate the validity of the framework, we employed two BCI testbeds, one for control and one for typing. We conducted experiments and compared the simulation results with the experimental outputs. The comparison between the results shows that simulation is sufficiently accurate in many cases. Even though simulation does not consider the effect of time-variant factors and cases of under- and over-estimation of performance are possible (as is the case with all simulations), our analysis showed that for many cases there is no statistically significant difference between the performance results of simulations and experiments. Overall, the simulation module is a useful tool to get a general idea about the upcoming experiment. In this manuscript, we have conducted a feasibility analysis for such a simulation approach; however, no parameters

were selected to be used in either free control or spelling experiments.

The current simulation framework assumes stationarity in the statistical characteristics of the EEG signals. However, EEG experiences nonstationarities for many different reasons such as user fatigue, background brain activity, external electronics, etc. Therefore, in our future work, to give a better representation of real experiments, we will develop a simulation module with nonstationary models for the class conditional distributions. Another limitation of the approach in this paper is the i.i.d. assumption of the EEG features in target and non-target classes. Especially non-target trials following a target are expected to have a different distribution due to the overlap with the ERPs corresponding to the target stimulus. Correspondingly, this phenomenon might cause simulations to overestimate the performance. For a better representation of the real operation, the simulations will be adapted to include a model which considers the overlap of the target trials.

Acknowledgements

This research is supported by NSF (CNS-1136027, IIS-1149570), NIH (2R01DC009834) and NIDRR (H133E140026). The authors acknowledge help and contributions from collaborators in the OHSU Reknew Projects Group and CSL at Northeastern.

ORCID

Mohammad Moghadamfalahi  <http://orcid.org/0000-0003-4525-9035>

References

- [1] Krusienski D, Sellers E, McFarland D, Vaughan T, Wolpaw J. Toward enhanced P300 speller performance. *J Neurosci Methods*. 2008;167:15–21.
- [2] Pfurtscheller G, Neuper C, Guger C, Harkam W, Ramoser H, Schlögl A, Obermaier B, Pregenzer M. Current trends in Graz brain-computer interface (BCI) research. *IEEE Trans Biomed Eng*. 2000;8:216–219.

- [3] Treder M, Blankertz B. (C) overt attention and visual speller design in an ERP-based brain-computer interface. *Behav Brain Funct.* 2010;6:28.
- [4] Arroyo S, Lesser RP, Poon W-T, Robert W, Webber S, Gordon B. Neuronal generators of visual evoked potentials in humans: visual processing in the human cortex. *Epilepsia.* 1997;38:600–661.
- [5] Bin G, Gao X, Wang Y, Hong B, Gao S. Vep-based brain-computer interfaces: time, frequency, and code modulations. [research frontier] *Comput Intell Mag IEEE.* 2009;4:22–26.
- [6] Nezamfar H, Mohseni Salehi SS, Erdogmus D. Stimuli with opponent colors and higher bit rate enable higher accuracy for c-vep bci. *Signal Processing in Medicine and Biology (SPMB), 2015 IEEE International Symposium.* 2015. p. 1–6.
- [7] Nezamfar H, Orhan U, Erdogmus D, Hild K, Purwar S, Oken B, Fried-Oken M. On visually evoked potentials in eeg induced by multiple pseudorandom binary sequences for brain computer interface design. *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference.* 2011;2044–2047.
- [8] Nezamfar H, Orhan U, Purwar S, Hild K, Oken B, Erdogmus D. Decoding of multichannel eeg activity from the visual cortex in response to pseudorandom binary sequences of visual stimuli. *Int J Imaging Syst Technol.* 2011;21:139–147.
- [9] Nezamfar H, Sinyukov D, Orhan U, Erdogmus D, Padir T. Brain interface to control a tele-operated robot. *Proceedings of the Fifth International Braincomputer Interface Meeting.* 2013.
- [10] Duda RO, Hart PE, Stork DG. *Pattern classification.* Chichester: John Wiley; 2012.
- [11] Orhan U, Erdogmus D, Roark B, Oken B, Fried-Oken M. Offline analysis of context contribution to erp-based typing bci performance. *J Neural Eng.* 2013;10:066003.
- [12] Orhan U, Hild K, Erdogmus D, Roark B, Oken B, Fried-Oken M. Rsvp keyboard: An eeg based typing interface. *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference.* 2012. p. 645–648.
- [13] Acqualagna L, Blankertz B. Gaze-independent bci-spelling using rapid serial visual presentation (rsvp). *Clin Neurophysiol.* 2013;124:901–908.
- [14] Acqualagnav L, Treder MS, Schreuder M, Blankertz B. A novel braincomputer interface based on the rapid serial visual presentation paradigm. *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE.* 2010;2686–2689.
- [15] Huang Y, Pavel M, Hild K, Erdogmus D, Mathan S. A hybrid generative/discriminative method for EEG evoked potential detection. *Neural Engineering, 2009. NER'09. 4th International IEEE/EMBS Conference.* 2009. p. 283–286.
- [16] Mathan S, Ververs P, Dorneich M, Whitlow S, Carciofini J, Erdogmus D, Pavel M, Huang C, Lan T, Adami A. Neurotechnology for image analysis: searching for needles in haystacks efficiently. *Augmented Cognition: Past, Present, and Future.* 2006.
- [17] Tangermann M, Hohne J, Stecher H, Schreuder M. No surprise-fixed sequence event-related potentials for brain-computer interfaces. *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE.* 2012. p. 2501–2504.
- [18] Friedman J. Regularized discriminant analysis. *J Am Stat Assoc.* 1989;84:165–175.
- [19] Akcakaya M, Peters B, Moghadamfalahi M, Mooney A, Orhan U, Oken B, Fried-Oken M. Noninvasive brain computer interfaces for augmentative and alternative communication. *IEEE Rev Biomed Eng.* 2014;7:31–49.
- [20] Hammon PS, de Sa VR. Preprocessing and meta-classification for braincomputer interfaces. *IEEE Trans Biomed Eng.* 2007;54:518–525.
- [21] Silverman B. *Density estimation for statistics and data analysis.* Chapman & Hall/CRC. 1998.
- [22] Orhan U. *RSVP Keyboard (TM): an EEG based BCI typing system with context information fusion (PhD thesis).* Boston (MA): Northeastern University.
- [23] Orhan U, Erdogmus D, Roark B, Purwar S, Hild K, Oken B, Nezamfar H, Fried-Oken M. Fusion with language models improves spelling accuracy for ERP-based brain computer interface spellers. *IEEE Engineering in Medicine and Biology Society Conference Proceedings.* 2011. p. 5774–5777.
- [24] Witten I, Bell T. The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Trans Inf Theory.* 1991;37:1085–1094.
- [25] Roark B, de Villiers J, Gibbons C, Fried-Oken M. Scanning methods and language modeling for binary switch typing. *Proceedings of the NAACL HLT 2010 Workshop on Speech and Language Processing for Assistive Technologies.* Association for Computational Linguistics. 2010. p. 28–36.
- [26] Oken BS, Orhan U, Roark B, Erdogmus D, Fowler A, Mooney A, Peters B, Miller M, Fried-Oken MB. Brain-computer interface with language model-electroencephalography fusion for locked-in syndrome. *Neurorehabil Neural Repair.* 2014;28:387–394.
- [27] MacKenzie IS, Soukoreff RW. Phrase sets for evaluating text entry techniques. *CHI'03 extended abstracts on Human factors in computing systems.* ACM; 2003. p. 754–755.
- [28] Townsend G, Shanahan J, Ryan DB, Sellers EW. A general p300 brain-computer interface presentation paradigm based on performance guided constraints. *Neurosci Lett.* 2012;531:63–68.
- [29] Serby H, Yom-Tov E, Inbar G. An improved P300-based braincomputer interface. *IEEE Trans Neural Syst Rehabil Eng.* 2005;13:89–98.
- [30] Treder M, Schmidt N, Blankertz B. Gaze-independent brain-computer interfaces based on covert attention and feature attention. *J Neural Eng.* 2011;8:066003.
- [31] Blankertz B, Krauledat M, Dornhege G, Williamson J, Murray-Smith R, Müller K-R. A note on brain actuated spelling with the berlin brain-computer interface. *Universal access in human-computer interaction. Ambient interaction.* New York, NY: Springer; 2007. p. 759–768.
- [32] Speier W, Arnold C, Deshpande A, Knall J, Pouratian N. Incorporating advanced language models into the p300 speller using particle filtering. *J Neural Eng.* 2015;12:046018.
- [33] Ryan D, Frye G, Townsend G, Berry D, Mesa-G S, Gates N, Sellers E. Predictive spelling with a p300-based brain-computer interface: increasing the rate of communication. *Int J Human-Comput Interact.* 2010;27:69–84.
- [34] Mainsah BO, Colwell KA, Collins LM, Throckmorton CS. Utilizing a language model to improve online dynamic data collection in p300 spellers. *IEEE Trans Neural Syst Rehabil Eng.* 2014;22:837–846.

- [35] Furdea A, Halder S, Krusienski D, Bross D, Nijboer F, Birbaumer N, Kübler A. An auditory oddball (p300) spelling system for brain-computer interfaces. *Psychophysiology*. 2009;46:617–625.
- [36] Riccio A, Mattia D, Simone L, Olivetti M, Cincotti F. Eye-gaze independent eeg-based brain-computer interfaces for communication. *J Neural Eng*. 2012;9:045001.
- [37] An X, Höhne J, Ming D, Blankertz B. Exploring combinations of auditory and visual stimuli for gaze-independent brain-computer interfaces. *PLoS One*. 2014;9(10).