

Increasing BCI communication rates with dynamic stopping towards more practical use: an ALS study

This content has been downloaded from IOPscience. Please scroll down to see the full text.

2015 J. Neural Eng. 12 016013

(<http://iopscience.iop.org/1741-2552/12/1/016013>)

View [the table of contents for this issue](#), or go to the [journal homepage](#) for more

Download details:

IP Address: 151.141.67.91

This content was downloaded on 27/01/2015 at 19:02

Please note that [terms and conditions apply](#).

Increasing BCI communication rates with dynamic stopping towards more practical use: an ALS study

B O Mainsah¹, L M Collins¹, K A Colwell¹, E W Sellers², D B Ryan²,
K Caves³ and C S Throckmorton¹

¹Duke University, Department of Electrical and Computer Engineering, USA

²East Tennessee State University, Department of Psychology, USA

³Duke University, Departments of Surgery, Medicine and Biomedical Engineering, USA

E-mail: leslie.collins@duke.edu

Received 3 September 2014, revised 13 November 2014

Accepted for publication 1 December 2014

Published 14 January 2015



Abstract

Objective. The P300 speller is a brain–computer interface (BCI) that can possibly restore communication abilities to individuals with severe neuromuscular disabilities, such as amyotrophic lateral sclerosis (ALS), by exploiting elicited brain signals in electroencephalography (EEG) data. However, accurate spelling with BCIs is slow due to the need to average data over multiple trials to increase the signal-to-noise ratio (SNR) of the elicited brain signals. Probabilistic approaches to dynamically control data collection have shown improved performance in non-disabled populations; however, validation of these approaches in a target BCI user population has not occurred. **Approach.** We have developed a data-driven algorithm for the P300 speller based on Bayesian inference that improves spelling time by adaptively selecting the number of trials based on the acute SNR of a user's EEG data. We further enhanced the algorithm by incorporating information about the user's language. In this current study, we test and validate the algorithms online in a target BCI user population, by comparing the performance of the dynamic stopping (DS) (or early stopping) algorithms against the current state-of-the-art method, static data collection, where the amount of data collected is fixed prior to online operation. **Main results.** Results from online testing of the DS algorithms in participants with ALS demonstrate a significant increase in communication rate as measured in bits/min (100–300%), and theoretical bit rate (100–550%), while maintaining selection accuracy. Participants also overwhelmingly preferred the DS algorithms. **Significance.** We have developed a viable BCI algorithm that has been tested in a target BCI population which has the potential for translation to improve BCI speller performance towards more practical use for communication.

Keywords: P300 speller, amyotrophic lateral sclerosis, Bayesian dynamic stopping

(Some figures may appear in colour only in the online journal)

1. Introduction

In recent years, brain–computer interfaces (BCIs) have received increased interest as alternative communication aids due to their potential to restore control/communication abilities to individuals with severe physical limitations due to neurologic diseases, stroke, and spinal cord injury [1, 2]. BCIs decode and translate brain electrical signals that convey

the user's intent into commands to control external devices such as a word spelling program for communication. People with amyotrophic lateral sclerosis (ALS), commonly known as Lou Gehrig's disease, represent a target population who could benefit from BCI system development [3]. ALS is a degenerative motor neuron disease that causes a progressive loss in voluntary muscle control. This often results in an inability to communicate either verbally or via gestures,

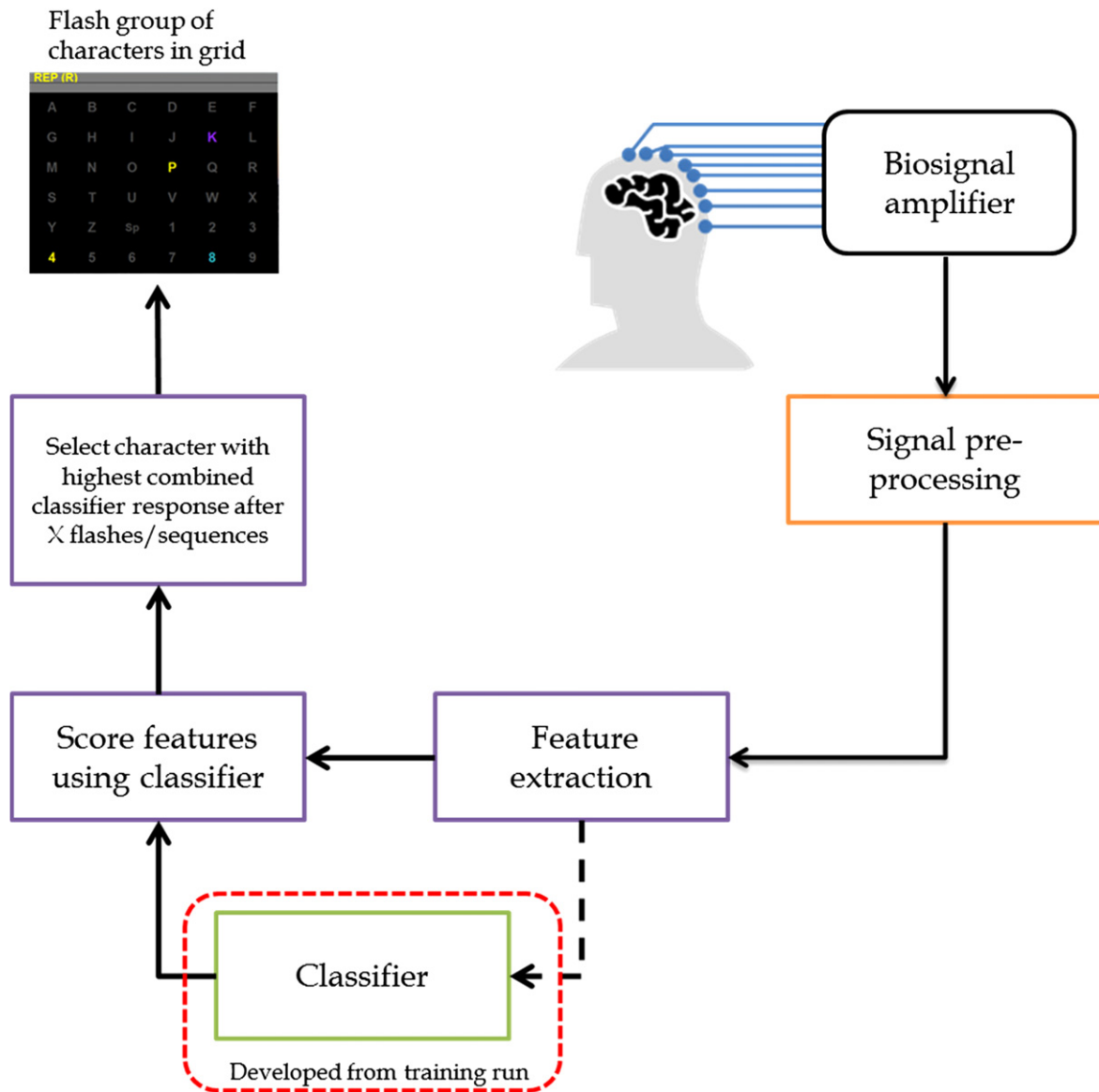


Figure 1. P300 speller components. The user is presented with a grid of character choices. External electrodes are used to measure EEG signals from the scalp, which are amplified, filtered and digitized for signal processing. From electrode channels, time sample blocks (800 ms) of EEG data following each flash are used to extract feature vectors to be used for classification. In the training run (broken arrow), feature vectors and their corresponding truth labels are used to train a classifier to distinguish between target ERP and non-ERP responses. During the test run, following each flash, feature vectors are extracted from post-stimulus time sample blocks of EEG data and scored with the trained classifier weights. The scores are averaged over multiple flashes/sequences and the character with the highest combined classifier response is selected as the target. The selected choice is presented as the user’s intended choice.

especially in the late stages of the disease, the ‘locked-in’ stage, where the loss of muscle control affects eye movements.

One of the most commonly researched BCI communication aids, especially in people with ALS, is the P300-based speller, developed by Farwell and Donchin [4], which enables users to make selections from an on-screen array by selecting a desired character or an icon that conveys a desired action. P300 spellers are BCIs that rely on eliciting event-related potentials (ERPs) in electroencephalography (EEG) data, including the P300 signal, which occurs in response to the presentation of rare stimuli within more frequent stimulus events in the context of an oddball paradigm [5]. Figure 1

shows a schematic of the P300 speller system. The user is presented with a grid of character choices and focuses on a desired character as groups of characters are flashed on the screen. The P300 speller operates by processing and analyzing a time window of EEG data after character subsets are flashed on the screen to discern the character that the user intended to spell. EEG features that correlate with the user’s intent are extracted from time windows of EEG data, and classification techniques [6, 7] are used to score features associated with each flash to distinguish between target ERP and non-target responses. This information is translated into a character selection on the screen after averaging data over multiple flashes.

In several studies, the P300 speller has been shown to be a viable system for communication in people with ALS [8–10], with potential stability for long-term use [11]. However, BCIs are still mainly used as research tools in controlled environments [12–14]. There are a limited number of BCI systems commercially available for independent home use, e.g. [15]. One of the primary reasons that BCIs have not translated into home use is that character selection times in BCI communication systems are very slow compared to other methods. Developing algorithms that improve the spelling rate has been a leading focus of BCI research in recent years.

While research suggests this system is a viable communication aid for people with ALS, the EEG responses from which P300 ERPs must be extracted have a very low signal-to-noise ratio (SNR). The standard approach to increase the P300 SNR within noisy EEG data is to average data from multiple trials for improved accuracy in classification performance [16]. This means each potential target character is sampled multiple times before the classifier determines the desired character, increasing the time needed to make a selection and decreasing the communication rate. In most conventional BCI spellers, the amount of data collected is fixed prior to online operation, this is termed *static data collection*, and the amount of data collected is usually similar across all users. Based on a literature search of P300 speller studies implemented in people with ALS using static data collection [3, 8–10, 14, 17–20], potential targets are flashed 8–40 times, with accuracies ranging from 60 to 100%, and performance improving in more recent years. However, static data collection, risks over or under collection of data since it does not assess the quality of the data that are measured.

Adaptive data collection strategies have been proposed in previous research to balance P300 speller accuracy and character selection time for improved online communication rate [21, 22]. Some approaches optimize the amount of fixed data collection prior to each P300 speller session by maximizing the written symbol rate metric [10], the number of selections a user can correctly make in a minute taking into account error correction e.g. [7, 10]. Most approaches vary the amount of data collection prior to each character selection based on a threshold function. Some approaches consider summative functions based on character P300 classifier scores e.g. [23–26]. Other approaches use a probabilistic model based on the P300 classifier scores, by maintaining a probability distribution over grid characters, updating the distribution following EEG data processing and stopping data collection when a specified confidence level is achieved [27–32]. Probabilistic data collection algorithms also provide a convenient framework to include additional knowledge such as *a priori* language information to further inform the algorithm's behavior without having to redesign the system.

Some adaptive data collection methods have relied on either tailoring the stopping criterion to a user's past performance (which may not be accurate longitudinally) or basing the stopping criterion on data collected from a pool of users. Relying on a pool of users to set the stopping criterion creates the potential for mismatch between the pool and new users. This is of special concern given that the pool of participants

has typically been healthy university students and employees [17] while the target BCI population is users with disabilities whose performance might differ substantially depending on the etiology and progression of their disabilities. Most P300 speller algorithm research is dominated by offline analysis or online testing in non-disabled participants, although there has been an increase in online validation in BCI target users in more recent years [3, 17, 33].

We have developed a data-driven Bayesian early stopping algorithm we term dynamic stopping (DS). Under DS, the algorithm determines the amount of data collection based on a confidence that a character is the correct target [34]. This enables flexibility in the amount of data collected based on the quality of the user's responses by collecting more data under low SNR conditions and less data under high SNR conditions without assuming a baseline level of performance for a particular user. Flash-to-flash assessments of EEG responses are integrated into the model via a Bayesian update of character probabilities. Data collection is stopped and a character is selected when the character probability exceeds a threshold. We further enhanced the algorithm by including information about the confidence of each character prior to data collection by exploiting the predictability of language via a statistical language model (dynamic stopping with language model—DSLML) [35]. In online studies of non-disabled participants, our algorithms significantly improved participant performance, with about 40% improvements in average bit rate from static to DS [34], and about 12% average improvements from DS to DSLML [35].

Similar DS algorithms with Bayesian approaches have been proposed, some including a language model, but have not been evaluated in a target BCI population. The goal of this study is to validate the improvements observed with our DS algorithms in the non-disabled user studies in a target BCI population. In this study, we compared the online performance of static and DS methods in participants with ALS.

2. Methods

2.1. Participants

The studies were approved by the Duke University Institutional Review Board and East Tennessee State University Institutional Review Board, respectively. Ten participants with ALS of varying impairment levels were recruited from across North Carolina and Tennessee by the Collins' Engineering Laboratory (Electrical and Computer Engineering Department, Duke University) and the Sellers' Laboratory (Psychology Department, East Tennessee State University), over a period of five months. Participants gave informed consent prior to participating in this study. Participant demographic information can be found in table 1. Participants who could not effectively communicate verbally used several forms of assistive technologies for communication, which included low and high tech approaches such lip reading, manual eye gaze boards, touchscreen devices and head

Table 1. Demographic information of study participants.

Participant	Age	Sex	ALSFRS-R score	Years post first symptoms	Inter-session intervals	Additional notes
D05	60	M	32	5.5	5 days, 7 days	- Normal head and eye movements - Severely impaired speech, communication via iPad (with text-to-speech software) and writing - Ambulatory
D06	59	M	3	7	8 days, 7 days	- No head movement, normal eye movements - No speech, communication via head tracking AAC system and manual eye-gaze board - Movement via wheelchair
D07	59	F	21	7	7 days, 8 days	- Normal head and eye movements - Severely impaired speech, communication via AAC device with mouse - Movement via wheelchair, ambulates short distances
D08	62	F	42	11	6 days, 1 day	- Normal head and eye movements - Slurred speech, communication via iPhone (with text-to-speech software) and writing - Ambulatory
E03	44	F	21	16	8 days, 7 days	- Normal eye movements - Moderately impaired speech - Movement via wheelchair
E20	38	M	5	8	28 days, 14 days	- Limited head movements, normal eye movements - No speech; communication via lip reading and eye gaze for directional attention; occasional use of head tracker system with predictive speller but cumbersome due to limited head/neck control - Movement via wheelchair
E21	63	F	1	8	28 days, 14 days	- Normal eye movements - No speech, communication via eye tracker system and manual eye gaze board - Movement via wheelchair
E23	57	M	30	2	3 days, 3 days	- Normal eye movements - Normal speech - Movement with walker
E24	56	M	33	3	4 days, 3 days,	- Normal eye movements - Normal speech - Movement via wheelchair
E25	49	M	33	1	14 days, 15 days	- Normal eye movements - Normal speech - Movement with a walker

Participants with 'D' were recruited at Duke University and those with 'E' were recruited at East Tennessee State University. ALSFRS-R denotes the 'ALS Functional Rating Scale' which provides a physician-generated estimate of the patient's degree of functional impairment, on a scale of 0 (high impairment) to 48 (low impairment) [47].

tracker systems. All of the recruited participants were used for data analysis.

2.2. P300 speller task

Participants were presented with a 6×6 P300 speller grid of alphanumeric characters on a screen, with the color checkerboard paradigm [10, 36] used for stimulus presentation. The speller grid consisted of alphanumeric characters, plus a space character. However, because this study involved single-word

spelling tasks, the space character was disabled, and if selected, was replaced with a hyphen (-). A word token was displayed on the top left corner of the grid, with the intended character to spell updated in parentheses displayed at the end of the word. The word tokens were randomly selected from the English Lexicon Project [37]. The task consisted of copy spelling the word by locating the target character in parentheses within the grid and counting the number of times it flashed. Groups of either three, four or five characters flashed

simultaneously on the screen. The flash duration and the inter-stimulus interval time were both set to 125 ms. Following each flash, a time window of EEG data was used to extract features to score each flash with a classifier. After a certain number of flashes, (fixed or dynamically determined), the P300 speller selects the character with the largest cumulative classifier response. After character selection, there was a 3.5 s pause prior to beginning selection of the next character.

Data were recorded from each participant in three sessions that were conducted on different days; each session consisted of a training and test run. Each session lasted about 1.5–2 h, including breaks. In the training runs, no feedback was presented to the user while labeled data was collected to train the P300 classifier. The training data for each participant consisted of three 6-lettered words. Features extracted from the training data were used to develop a P300 classifier that was used in all the test runs for each session. In the test runs, using the trained classifier, participants performed the copy-spelling task with feedback and no error correction. During each test run, the three data collection algorithms, static, DS and DSLM, were used. The algorithm order was randomized with each test run to avoid biasing the results by algorithm order. The testing data for each participant consisted of two 6-lettered words per algorithm for a total of 36 characters per algorithm across all three sessions. Prior to the DS algorithms, participants were told to expect variable data collection in order to minimize surprise or confusion.

2.3. Signal acquisition

The open source BCI2000 software was used for this study [38]. Additional functionality was added to implement the DS algorithms [34, 35]. The Collins' Lab collected EEG signals using 32-channel electrode caps in the clinic; the Sellers' Lab used 16-channel electrode caps and collected data in participants' homes. EEG signals were connected to the computer via gUSBamp biosignal amplifiers. The left and right mastoids were used for ground and reference electrodes, respectively. The signals were filtered (0.1–60 Hz at Duke University, 0.5–30 Hz at ETSU) and digitized at a rate of 256 samples/s. Data collected from electrodes Fz, Cz, P3, Pz, P4, PO7, PO8, and Oz were used for signal processing [6].

2.4. Feature extraction and P300 classifier training

Features that correlate with the user's intent were extracted from the EEG training data and used to develop a participant-specific P300 classifier, according to Krusienski *et al* [6]. Following each flash, a time window of 800 ms of EEG data (205 samples at sampling rate of 256 Hz), was obtained from each of the 8 electrode channels and down sampled to a rate of 20 Hz by averaging 13 time samples to obtain a feature point (the last ten were not used for computational simplicity of matrix operations). The time averaged features were concatenated across the channels to obtain a feature vector, $\mathbf{x} \in \mathbb{R}^{1 \times 120}$, per flash i.e. 15 features/channel \times 8 channels/flash = 120 features/flash. A truth label was assigned to a flash, $t \in \{0, 1\}$, depending on whether the target character

was present in the flash, i.e. label '1' if the target character was present in the flash and label '0' if the target character was not present in the flash.

A training data set from a session for each participant consisted of feature vectors and their corresponding truth labels, $\mathbf{T} = \{(\mathbf{x}_1, t_1), \dots, (\mathbf{x}_T, t_T)\}$. The training dataset was used to train a stepwise linear discriminant analysis (SWLDA) classifier, $\mathbf{w} \in \mathbb{R}^{1 \times 120}$. SWLDA uses a combination of forward and backward ordinary least-squares regression steps to add and remove features based on their ability to discriminate between classes. The *p-to-enter* and *p-to-remove* were set to 0.10 and 0.15, respectively.

2.5. Data collection stopping criteria

During the test run, character selection was made after analyzing EEG data from multiple flashes. Following each flash, i , a feature vector, \mathbf{x}_i , is extracted and used to compute a flash score with the trained classifier, $y_i = \mathbf{w}\mathbf{x}_i^T$. A sequence is a unit of data collection which consists of flashing all the defined character flash sub-sets once. For the checkerboard paradigm using the 6×6 speller grid (shown in figure 1), a sequence consists of 18 flashes, with each character flashed twice per sequence.

2.5.1. Static stopping. In static stopping, all characters in the grid, $C_n \in \mathbf{G}$, begin data collection with a zero score. With each group of characters that are flashed on the screen, $C_n \in \mathbf{S}_i$, their scores are updated by adding the computed flash score, y_i . After a fixed number of sequences, the character with the maximum cumulative classifier score was selected as the target character. A character's classifier score is updated according to:

$$\zeta(C_n) = \sum_{i=0}^I \mu_{i,n} y_i, \quad (1)$$

where $\zeta(C_n)$ is the cumulative classifier for the character, C_n after I flashes prior to character selection; y_i is the classifier score for the i th flash; $\mu_{i,n} = 1$ if $C_n \in \mathbf{S}_i$; and $\mu_{i,n} = 0$ if $C_n \notin \mathbf{S}_i$. For this study, character selection for static stopping was made after 7 sequences i.e. $I = 18 \times 7 = 126$ flashes.

2.5.2. Bayesian DS. In DS, the number of flashes prior to character selection was determined by updating the character probabilities of being the target after each flash via Bayesian inference and stopping when a threshold probability is attained. The DS algorithm [34], consists of an online and offline portion, figure 2. In the offline portion, the trained classifier is used to score the EEG data from the training session. The scores are grouped into non-target and target EEG response scores. The histograms of the grouped classifier scores are scaled and then smoothed with kernel density estimation to generate likelihood probability density functions (pdfs) for the target, $p(y_i|H_1)$, and non-target, $p(y_i|H_0)$, responses.

In the online portion, the pdfs are used in the Bayesian update process. Prior to spelling a new character, characters

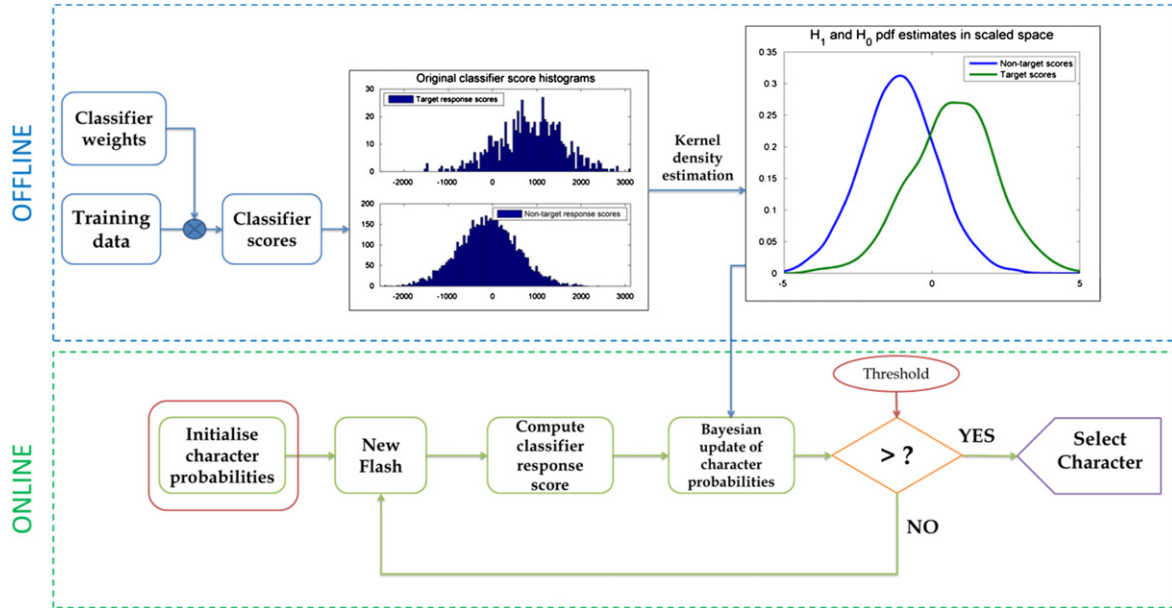


Figure 2. Flowchart of Bayesian dynamic stopping algorithm [34]. In the offline portion (top blue panel), EEG data from a participant training session are grouped into target and non-target responses and classifier scores are calculated using the SWLDA classifier weights. Kernel density estimation is used to smooth the histograms of the grouped scores to generate likelihood pdfs. In the online portion, character probabilities are initialized either from a uniform distribution or a language model. With each new flash, character probabilities are updated with Bayesian inference until a threshold probability is met.

are each assigned an initialization probability, $P(C_n = C_T^* | \mathbf{Y}_0)$. All characters begin data collection with an equal likelihood of being the target character, i.e. $P(C_n = C_T^* | \mathbf{Y}_0) = \frac{1}{N}$, where N is the total number of choices in the grid. With each new flash, the character probabilities are updated via Bayesian inference:

$$P(C_n = C_T^* | \mathbf{Y}_i, \mathbf{S}_i) = \frac{p(y_i | C_n = C_T^*, \mathbf{S}_i) \times P(C_n = C_T^* | \mathbf{Y}_{i-1}, \mathbf{S}_{i-1})}{\sum_{j=1}^N p(y_i | C_j = C_T^*, \mathbf{S}_i) \times P(C_j = C_T^* | \mathbf{Y}_{i-1}, \mathbf{S}_{i-1})} \quad (2)$$

where $P(C_n = C_T^* | \mathbf{Y}_i, \mathbf{S}_i)$ is the posterior probability of the character C_n being the target character, C_T^* , given the current classifier score history, $\mathbf{Y}_i = [y_1, \dots, y_{i-1}, y_i]$, and the current subset of flashed characters, \mathbf{S}_i ; $p(y_i | C_n = C_T^*, \mathbf{S}_i)$ is the likelihood of the classifier score, y_i , given that the character C_n was present or not present in the flashed subset, \mathbf{S}_i ; $P(C_n = C_T^* | \mathbf{Y}_{i-1}, \mathbf{S}_{i-1})$ is the prior probability of the character C_n being the target character; and the denominator normalizes the probabilities over all characters. The likelihood $p(y_i | C_n = C_T^*, \mathbf{S}_i)$ is assigned depending on whether C_n was/was not present in the subset of flashed characters, \mathbf{S}_i :

$$p(y_i | C_n = C_T^*, \mathbf{S}_i) = \begin{cases} p(y_i | H_1), & \text{if } C_n \in \mathbf{S}_i, \\ p(y_i | H_0), & \text{if } C_n \notin \mathbf{S}_i. \end{cases} \quad (3)$$

Figure 3 shows the evolution of character probabilities during the Bayesian character probabilities with each new flash. Data collection is stopped when a character probability attains the threshold probability, set at 0.9, and the character

is selected as the target character. A minimum limit on the amount of data collection was not imposed. However, a maximum limit of 7 sequences was imposed (as convergence is not guaranteed), and if the threshold probability is not reached, the character with the maximum probability is selected as the target character. For the next character, character probabilities are re-initialized and the Bayesian update process is repeated until character selection.

2.5.3. Bayesian DSLM. In dynamic stopping with a language model, the Bayesian update process is identical but for the initialization probabilities, $P(C_n = C_T^* | \mathbf{Y}_0)$, which are dependent on the previous character selection, a bigram language model [35]. For simplicity, we initialized the probabilities of the first character assuming each character is equally likely to be the target character; alternatively a language model could be used based on the probability of being a first letter in a word. For subsequent character selection, if the previous character selection was non-alphabetic, the initialization probabilities were uniform. If the previous character was non-alphabetic, a letter bigram model was used, shown in figure 4, generated from the Carnegie Mellon University online dictionary [39]. Element $(i_{\text{row}}, j_{\text{column}})$ in the character probability matrix denotes the conditional probability, $P(\mathcal{A}_j | \mathcal{A}_i)$, that the next letter is the j th letter of the alphabet, given the i th letter was previously spelled. For example, vowels are more likely to follow consonants, and specifically, a ‘U’ is more likely to follow a ‘Q’, compared to an ‘E’ or ‘S’.

For non-alphabetic characters (NACs), the initialization probabilities are set to $\frac{1}{N}$. For alphabetic characters, the initialization probabilities are dependent on the previous letter

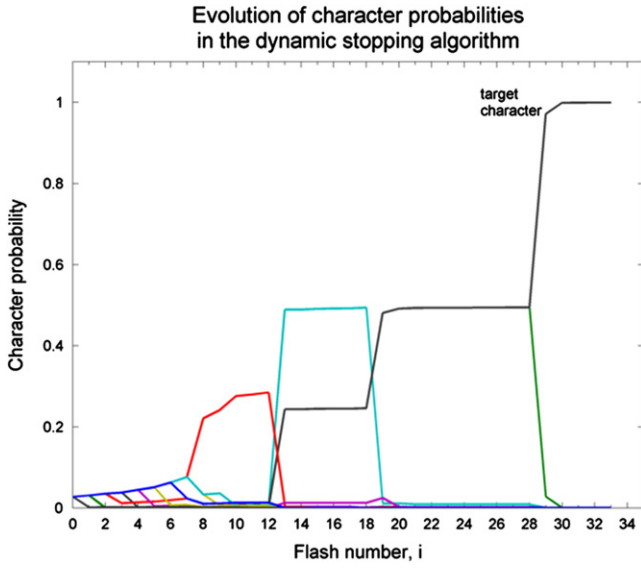


Figure 3. Evolution of character probabilities in the dynamic stopping algorithm. Prior to data collection, character probabilities are initialized. With each new flash, character probabilities are updated via Bayesian inference. After some flashes, the character probability distribution becomes sparser as a few likely target characters start to emerge. However, the probability mass starts to concentrate on one character, and ideally, the probability of the target character should converge towards 1. Data collection is stopped when a character’s probability attains a preset threshold value, and it is selected as the target character.

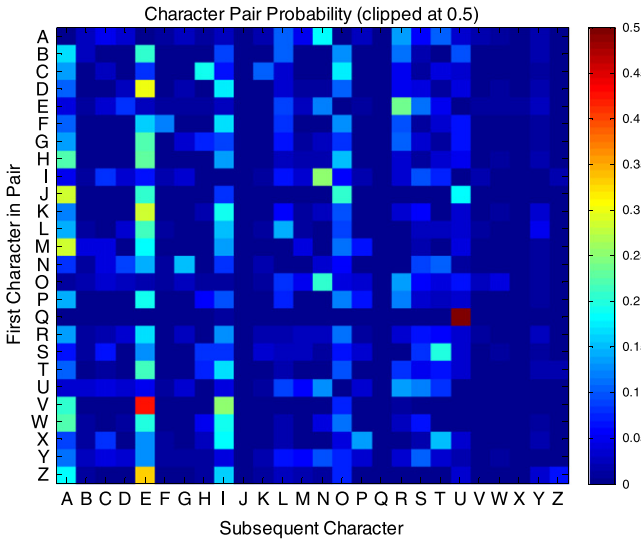


Figure 4. Character probability matrix for bigram language model. Probability matrix was developed from Carnegie Mellon University dictionary [39]. Element $(i_{\text{row}}, j_{\text{column}})$ in the grid denotes the conditional probability, $P(A_j|A_i)$, that the user will spell the j th letter in the alphabet, A_j , given the i th letter, A_i , was previously spelled. Probability values are clipped at 0.5 for visualization.

selection, \mathcal{A}_{T-1} , consisting of a weighted average of a bigram language model and a uniform distribution:

$$P(C_n = C_T^* | Y_0) = \frac{1}{N}, C_n \notin \{A: Z\}, \quad (4)$$

$$P(C_n = C_T^* | Y_0) = \alpha P(\mathcal{A}_T = C_n | \mathcal{A}_{T-1}) \left(1 - \sum_{\text{NAC}} \frac{1}{N}\right) + (1 - \alpha) \frac{1}{N}, C_n \in \{A: Z\}, \quad (5)$$

where α denotes the weight of the language model; $P(\mathcal{A}_T = C_n | \mathcal{A}_{T-1})$ denotes the conditional probability that the next letter is C_n given the previously spelled letter is \mathcal{A}_{T-1} , obtained from the character probability matrix; $\sum_{\text{NAC}} \frac{1}{N}$ is the sum of all NACs, subtracted from 1 to normalize the probabilities; $1 - \alpha$ is the weight of the uniform distribution. The additional probability introduced by the uniform distribution is an error factor to mitigate the possible influence of incorrectly spelled characters on the initialization probabilities. Based on offline simulations, the weight of the language model was set to $\alpha = 0.9$.

2.6. Performance measures

The online performances of the participants using the P300 spellers with the various data collection algorithms were evaluated using accuracy, task completion time, and communication rates. All performance measures were pooled across the three sessions. The accuracy is the percent of characters correctly spelled by the participant. The task completion time is the time spent to complete the task, determined from the total number of flashes to complete the spelling task, including the time pauses between flashes:

$$CST_n = 3.5 + (F_n * (ISI + FD)) s, \quad (6)$$

$$TCT = \frac{\sum_{n=1}^{36} CST_n}{60} \text{ min}, \quad (7)$$

where CST_n is the character selection time for character C_n ; 3.5 s is the time pause between character selections; F_n is the number of flashes used to select C_n , ISI is the inter-stimulus interval and FD is the flash duration; and TCT is the task completion time.

Bit rate is a communication measure that takes into account accuracy, task completion time and the number of choices in the grid [40]:

$$B = \log_2 N + P \log_2 P + (1 - P) \log_2 \left(\frac{1 - P}{N - 1} \right), \quad (8)$$

$$\text{Bit rate} = B * \frac{\text{total number of characters spelled}}{\text{TCT}}, \quad (9)$$

where B is the number of transmitted bits/character selection; N is the number of possible character selections in the speller grid; P is the participant accuracy. Theoretical bit rate was also calculated and differs from bit rate by excluding the time pauses between character selections. Since the time pauses can be varied according to user comfort, theoretical bit rate represents an upper bound on possible communication rate. However, in practice, some pause between characters is required for the BCI user to evaluate the feedback provided by the BCI.

Statistical analyses of the data involved repeated measures ANOVA tests to analyze the effect of the various data collection algorithms, followed by multiple comparison tests with Bonferroni adjustments, for pairwise comparisons if applicable ($p < 0.05$).

3. Results

Figure 5 shows the participant non-target and P300 ERPs for electrode C_z , averaged across all three sessions. Most participants were able to elicit a P300 response for BCI control, with some participants having relatively strong ERP responses e.g. D05 and D08. Average participant performance was calculated by pooling across all P300 speller sessions. Statistical tests revealed significant differences in the mean of at least two algorithms for three of the four performance measures: task completion time, bit rate and theoretical bit rate. Additional multiple comparison tests were done to determine specifically which pairs of algorithms with significant differences. The same trend was observed across the performance measures that had significant differences: static data collection was significantly different from both DS and DSLM, however, there was no significant difference between DS and DSLM.

Figure 6 shows the mean, minimum and maximum number of flashes used for character selection for each participant under each of the algorithms. In general, the higher the participant performance level, the fewer flashes required prior to character selection. However, a wide range in the number of flashes per character selection with the DS algorithms occurs for each participant. This demonstrates that choosing a static amount for data collection based on a user's performance level may not consistently match data collection needs. The range in the number of flashes within subjects demonstrates how the DS algorithms adapt to acute changes in user performance rather than relying on an arbitrary number of flashes or the user's past performance.

The total task completion time was determined from the total number of flashes used to spell all words, including the time pauses between character selections. The static completion time was the same for all participants (35 s per character, 21 min per task), although one participant (E21) selected one less character resulting in a slightly lower time. Figure 7(A) shows the total task completion time and average amount of time required to select a character. Character selection times were significantly less in both DS algorithms, with most participants achieving a 45–75% reduction when compared to the static stopping algorithm ($p < 10^{-6}$). The DS algorithms reduced character selection time, with most ranging from 8 to 20 s/character. Figure 7(B) shows that average participant accuracy decreased from static stopping ($79.44 \pm 29.98\%$), but the differences were not significant between any of the algorithms, DS ($75.40 \pm 27.16\%$) and DSLM ($76.39 \pm 25.63\%$) ($p < 0.23$). Despite the substantial reduction in data collected with DS, no significant deterioration in accuracy was observed.

The accuracy and task completion time were used to calculate the bit rate and theoretical bit rate. Figures 7(C)–(D) show that the communication rates remained the same or improved from the static to the DS conditions. Figures 7(C) and (D) reveal that most participants obtained a substantial increase in their bit rates (100–300%) and theoretical bit rates (100–650%) with the DS algorithms, due to maintaining similar accuracy levels while significantly reducing the spelling task completion time. There was a significant increase in bit rate from static to DS and DSLM ($p < 10^{-3}$) and in theoretical bit rate from static to DS and DSLM ($p < 10^{-6}$). There was no significant difference observed in communication rate between the DS algorithms. The average participant results are summarized in table 2.

Testing BCI algorithms in the target population serves several important purposes. Results from non-disabled participants may not necessarily translate to the disabled population due to variability in disease cause and progression [17]; thus, testing in the target population is a key validation step for an algorithm. Further, testing in the target population presents an opportunity for useful feedback during the design process from target users [33]. From the post-session survey results shown in figure 8, we observed that despite varying accuracy levels, participant algorithm preference significantly increased from static, to DS, to DSLM algorithm ($p < 10^{-5}$). It should be noted that we informed participants to expect variable data collection with the DS algorithms in order to minimize surprise or confusion.

Given that multiple sessions occurred for data collection, it is of interest to consider whether the classifiers trained in each session were similar. However, a direct comparison of the weight vectors that define the classifiers is difficult due to the variation in EEG data and the sparsity constraint imposed by SWLDA. As an alternative approach, classifiers were applied offline to test data collected in separate sessions. If the classifiers provide consistent performance across sessions, it might be assumed that the data collected for calibration is fairly stable. For each session, classifiers trained from the other two sessions were used to simulate P300 speller selections with EEG data from the static stopping algorithm run. Participant session intervals varied from 3 days to about 1 month (see table 1). Figure 9 shows the average offline accuracy compared with the average online accuracy using the within-session classifier. It can be observed that participant within-session accuracy was comparable to the inter-session accuracy. These results indicate there is potential to reduce training time with this *a priori* information, possibly by initializing and updating a classifier as training data is collected.

4. Discussion

Our DS algorithms adapt the amount of data collection based on acute changes in user performance to maximize spelling speed without compromising accuracy, compared to static stopping. Participants demonstrated a range in the number of

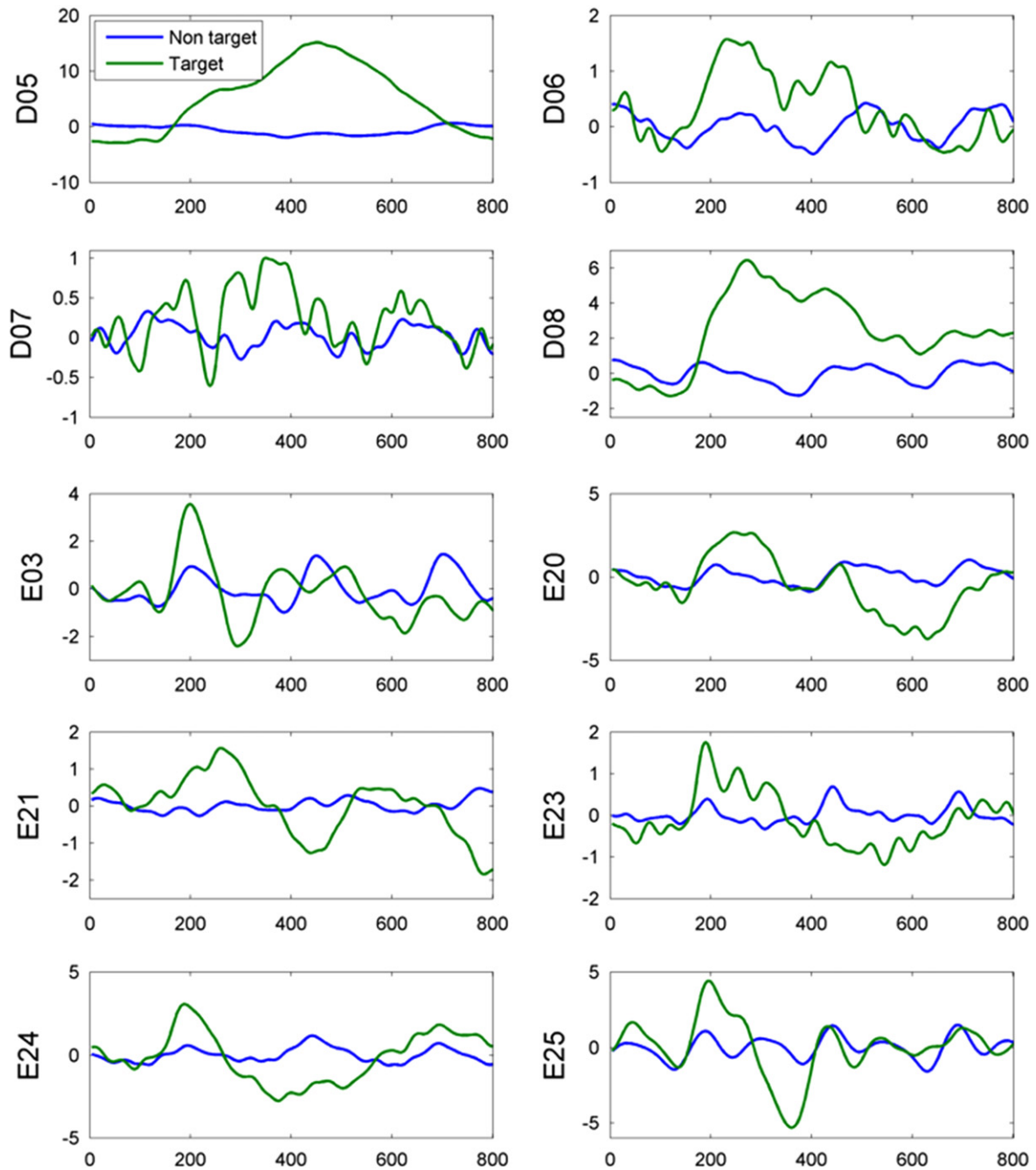


Figure 5. Average participant ERP signals for electrode Cz. The average waveforms were obtained from training data across the three sessions.

flashes to select each character, indicating that performance even within a participant is not constant. This highlights the importance of not relying on an arbitrary number of flashes or the user's past performance to control data collection. Most participants experienced a significant reduction in task completion times with the DS algorithms with little to no negative impact on accuracy. Further, communication rates were greatly improved for the majority of participants.

In contrast to our non-disabled studies where significant performance improvements were observed from static to DS [34], and from DS to DSLM [35], the inclusion of a language model did not significantly improve the performance of the

DS algorithm with the ALS participants. There is likely more variability in the participants with ALS population due to other confounding factors such as differences in disease etiology and progression, in contrast to the non-disabled population which tends to be more uniform and skewed towards a younger demographic. However, the inclusion of a language model did not significantly cause a reduction in accuracy despite the substantial reduction in data collection. Some previous approaches that include language information in BCI spellers involved changing the user-interface [41–43]. Sometimes this might negatively affect accuracy, as in Ryan *et al* [42], where it was hypothesized that the negative impact

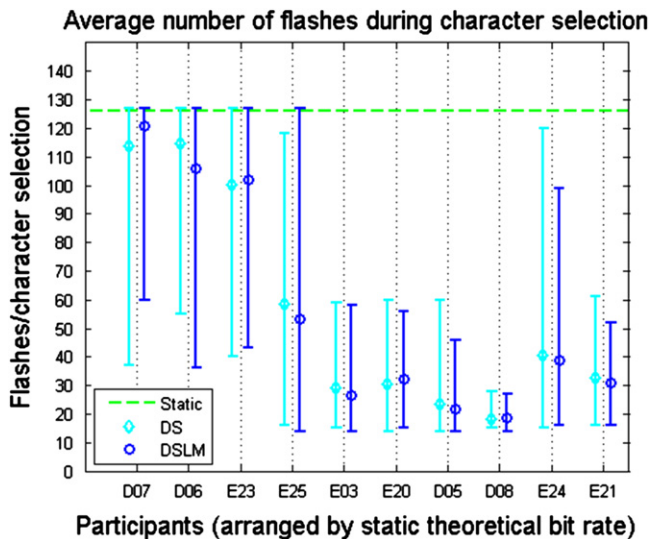


Figure 6. Average number of flashes per character selection with static stopping, dynamic stopping (DS) and dynamic stopping with language model (DSLML) algorithms. The error bars indicate the maximum and minimum number of flashes/character selection used by each participant. There was no minimum number of flashes imposed. The maximum number of flashes possible to spell each character was 126 flashes due to a sequence limit of 7 sequences/character with 18 flashes/sequence using the checkerboard paradigm on a 6×6 P300 speller grid.

was likely due to the increased task difficulty of adapting to a new interface layout with multiple selection windows. In our algorithm, the language model is incorporated within the data collection algorithm, leaving the speller interface unchanged. While this method of incorporating a language model did not result in significant performance improvements, the language model used was a bigram model which is overly simplistic. A more complex language model that takes into account all of the previously spelled characters could be incorporated in a similar manner, as could other natural language processing techniques, potentially improving performance.

Several DS algorithms for ERP-based spellers have been proposed in the literature [22], and we focus on those that have been implemented online, as real-time closed-loop online BCI feedback gives a better measure of the performance of an algorithm compared to offline analysis. Some DS approaches have utilized summative functions based on character P300 classifier scores [23–26]. However, these approaches relied on estimating threshold parameters based on a participant pool average or using the same parameters across users. A user's performance can vary acutely based on signal SNR, artifact level, attention, mood etc [44, 45] and there is inter-participant variability in performance [46]. Alternatively, a probabilistic-based stopping criterion allows for a more flexible means to adapt data collection based on changes in a user's acute performance as this uncertainty is captured via classifier score distributions. Other DS algorithms have used a probabilistic model, similar to the one in this study, where character probability values are initialized, updated with information derived with additional EEG data collection until a preset probabilistic threshold level is met

and the character with the maximum *a posteriori* probability is selected [27–30]. However, these algorithms were tested with offline simulations, followed by online studies in non-disabled participants and require further validation in a target BCI population.

Online testing in people from a target BCI population is a key step in BCI algorithm development because algorithms optimized for individuals without disability may not necessarily generalize to the target BCI population. Our algorithm development process included offline simulations, testing in participants without disability, followed by online validation with participants with ALS. The positive feedback from the people with ALS provides an incentive to incorporate our adaptive data collection algorithm in prototype BCI speller systems for home use, given the significant improvements obtained when compared to the current state-of-the-art static stopping method. In addition, this reinforces the need for developing BCI systems with communication rates that are comparable to other augmentative and assisted communication systems to translate BCIs into practical systems for daily home use.

There appears to be no correlation between level of impairment and BCI control. ALSFRS-R denotes the revised ALS Functional Rating Scale [47], which provides a physician-generated estimate of the patient's degree of functional impairment, on a scale of 0 (high impairment) to 48 (low impairment) (table 1). The ALSFRS-R scores of the low-performing participants D06, D07 and E23, were 3, 21 and 30, respectively. The remaining participants who had varying levels of impairment ranging from 1 to 42, performed with accuracy levels in the 85–100% range. In McCane *et al* where participants with ALS performed word spelling tasks with a P300 speller, results suggest no correlation between level of impairment and successful BCI use [48]. Most of the high performing participants (>70% accuracy) had ALSFRS-R scores of less than 5, with the rest ranging from 16 to 25. The low performing participants (<40% accuracy) had some visual impairment e.g. double vision, rapid involuntary eye movements and drooping eyelids, which could hinder their ability to focus on the target for effective BCI use [49]. All participants in this study had the ability to control eye gaze. One possible reason for poor performance is the ability to elicit P300 ERPs to control BCIs as it can be observed from figure 5 that the low performing participants elicited P300 ERPs with relatively low SNR compared to non-target responses. However, other possible reasons for poor performance could include presence of artifact or the user misunderstanding instructions for BCI use [50]

Head and eye tracking systems are commercially available and can provide a convenient means of daily communication for people with severe neuromuscular ability e.g. eye-gaze calibration time can take around 3–5 min. Setting-up, calibrating and troubleshooting a BCI system may be difficult for lay people such as family members or care-takers with limited technical background, especially given the day-to-day variability of user performance, environmental conditions, etc. Nonetheless, a recent case study by Sellers *et al* showed that a stroke survivor was not able to accurately use

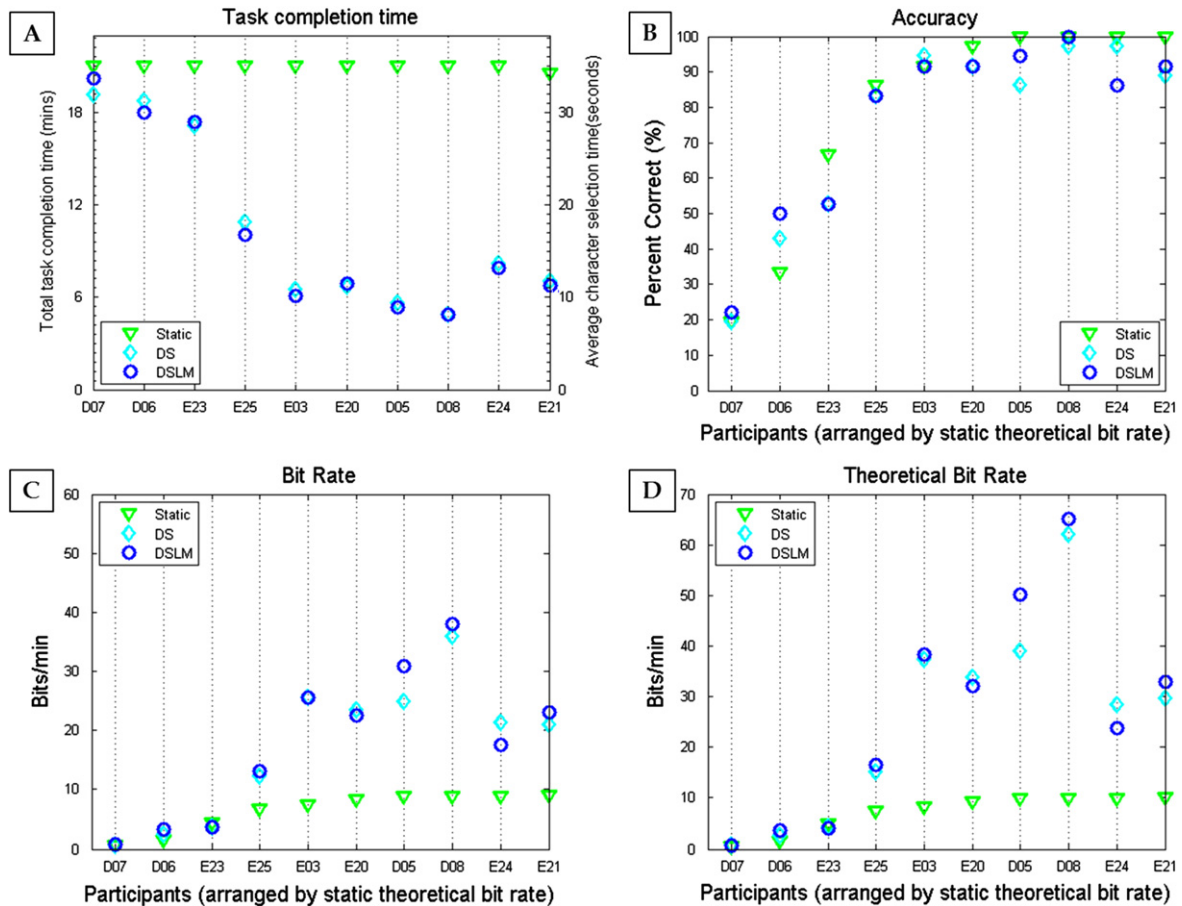


Figure 7. Comparison of performance measures between static, dynamic stopping (DS) and dynamic stopping with language model (DSLML) for (A) task completion time, (B) accuracy, (C) bit rate, and (D) theoretical bit rate in ALS patient study. The accuracy is the percentage of characters correctly spelled by the user across test sessions. The maximum possible completion time, with pauses (3.5 s) between character selections, was 21 min due to a sequence limit of 7. Bit rate is a communication rate that takes into account accuracy, task completion time and the number of possible character choices of a communication channel. Theoretical bit rate excludes the time pauses between character selections and represents an upper bound on the user’s possible communication rate.

Table 2. Summary of performance measures comparing static, dynamic stopping (DS) and dynamic stopping with language model (DSLML) conditions.

Average performance measure	Static	DS	DSLML	<i>p</i> -value
Time to complete task (min)	21.00 *▼	10.47 ± 5.69*	10.37 ± 5.86▼	<9 × 10 ⁻⁷
Accuracy (%)	79.44 ± 29.98	75.40 ± 27.16	76.39 ± 25.63	<0.23
Bit rate (bits/min)	6.44 ± 3.21*▼	17.06 ± 11.78*	25.22 ± 19.56▼	<3.15 × 10 ⁻⁴
Theoretical bit rate (bits/min)	7.13 ± 3.56*▼	17.82 ± 15.54*	26.71 ± 21.21▼	<7.1 × 10 ⁻⁷

- Repeated measures ANOVA was used to determine differences between algorithm means at 5% level of significance, i.e. a *p*-value <0.05 indicates at least two means are significantly different. When applicable, post-hoc multiple comparison tests (with Bonferonni adjustment) were performed to determine specifically the algorithms for which significant differences occurred.
 - Symbols * or ▼ indicate pairs with significant differences.

an eye tracking system, but was successful using a matrix speller, obtaining accuracy levels >70% [51]. BCI systems can potentially be a viable communication alternative when eye-tracker systems fail to provide or does not provide effective communication, especially in late-stage ALS where there can be an inability to sustain controlled eye movements.

Our Bayesian DS algorithm requires further development prior to being adapted for translational purposes. The spelling task was designed with no error correction to test the

robustness of the error factor in the DS algorithm. Correcting selection errors requires at least two selective actions: deleting the erroneous character and reselecting the correct character. It has been recommended that P300 spellers perform with accuracy levels >70% for practical communication, to account for erroneous character revisions [52], and thus participants D07, D06 and E23 may have accuracy levels that are too low for effective communication. Improving the accuracy levels of low performing users is a key next step which

SURVEY QUESTIONS

1. The set of words that were easiest to spell
2. The set of words that were quickest to spell
3. The set of words that required the least amount of mental effort to spell
4. The condition you preferred

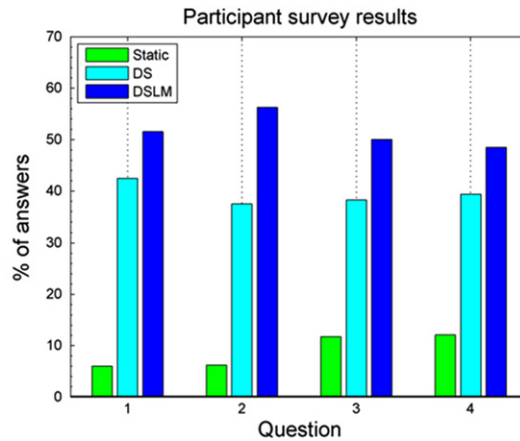


Figure 8. Survey results of algorithm preference for ALS study. The survey questions were asked of each of the ALS participants after each P300 speller test session in which the test order of the three algorithms (Static, DS, and DSLM) was randomized.

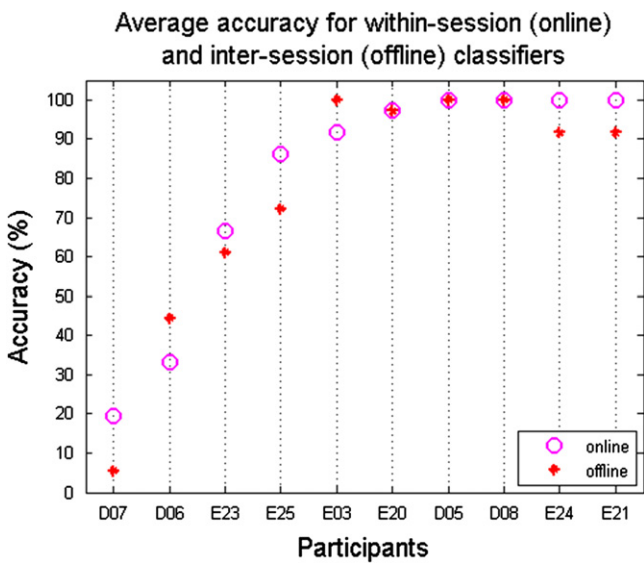


Figure 9. Session classifier comparison for static stopping. The average accuracy measured online shows the average within-session classifier performance across all three sessions. Offline analysis was used to compute the average accuracy for the inter-session classifiers.

requires further P300 classifier development, potentially through more sophisticated language models, natural language processing tools like dictionary-based spelling correction [53], or predictive text. While language models and spelling correction can be incorporated into the algorithm without impact on the speller interface, predictive text must be integrated into the BCI system while taking into account the limited ability of target BCI users to navigate choices. One possibility is to modify the algorithm to include predictive word options incorporated directly into the speller grid, as in Kaufmann *et al* [43], to minimize spelling task difficulty as word selection occurs in a similar manner to character-based selection. Finally, the complication of spelling phrases or sentences, where word-space boundaries are important, must be considered. The spelling task designed in this study, like most P300-based BCI studies, involved single-

word copy-spelling tasks where word length is known *a priori*; however, a more realistic system would test the algorithm on phrases or sentences. This will likely require additional consideration with the language model as to when to transition from word to space/punctuation to word boundaries, or how robust the system will be to errors and still be able to discern the user’s intended message. Further development of the language model, automatic error correction, and predictive text will likely lead to speller performance improvements and further the transition of the system from research lab to home.

5. Conclusion

Overall, we provide a viable BCI algorithm that has been validated in target BCI users with results indicating the potential advantage of using an adaptive data collection to improve P300 speller efficiency. Our adaptive algorithm has potential for translation into prototype BCI speller systems given the significant performance improvements over the current state-of-art data collection method and the positive feedback from the participants with ALS.

Acknowledgments

This research was supported by NIH/NIDCD grant number R33 DC010470-03. The authors would like to thank the participants for dedicating their time to this study. The authors would also like to thank Ken Kingery (Duke University, Durham NC 27708, USA) for editing the manuscript.

References

- [1] Wolpaw J and Wolpaw E W 2012 *Brain–Computer Interfaces: Principles and Practice* (New York: Oxford University)
- [2] Akcakaya M, Peters B, Moghadamfalahi M, Mooney A R, Orhan U, Oken B, Erdogmus D and Fried-Oken M 2013 Noninvasive brain–computer interfaces for augmentative and alternative communication *Biomed. Eng.* **7** 31–49

- [3] Moghimi S, Kushki A, Guerguerian A M and Chau T 2013 A review of EEG-based brain-computer interfaces as access pathways for individuals with severe disabilities *Assist. Technol.* **25** 99–110
- [4] Farwell L A and Donchin E 1988 Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials *Electroencephalogr. Clin. Neurophysiol.* **70** 510–23
- [5] Sutton S, Braren M, Zubin J and John E R 1965 Evoked-potential correlates of stimulus uncertainty *Science* **150** 1187–8
- [6] Krusienski D J et al 2006 A comparison of classification techniques for the P300 Speller *J. Neural Eng.* **3** 299–305
- [7] Aloise F, Schettini F, Arico P, Salinari S, Babiloni F and Cincotti F 2012 A comparison of classification techniques for a gaze-independent P300-based brain-computer interface *J. Neural Eng.* **9** 045012
- [8] Sellers E W and Donchin E A 2006 P300-based brain-computer interface: initial tests by ALS patients *Clin. Neurophysiol.* **117** 538–48
- [9] Hoffmann U, Vesin J-M, Ebrahimi T and Diserens K 2008 An efficient P300-based brain-computer interface for disabled subjects *J. Neurosci. Methods* **167** 115–25
- [10] Townsend G et al 2010 A novel P300-based brain-computer interface stimulus presentation paradigm: moving beyond rows and columns. *Clin. Neurophysiol.* **121** 1109–20
- [11] Sellers E W, Vaughan T M and Wolpaw J 2010 A brain-computer interface for long-term independent home use *Amyotrophic Lateral Sclerosis* **11** 449–55
- [12] Berger T W 2008 *Brain-Computer Interfaces: An International Assessment of Research and Development Trends* (Berlin: Springer) doi:10.1007/978-1-4020-8705-9
- [13] Brunner P, Bianchi L, Guger C, Cincotti F and Schalk G 2011 Current trends in hardware and software for brain-computer interfaces (BCIs) *J. Neural Eng.* **8** 025001
- [14] Sellers E W, McFarland D J, Vaughan T M and Wolpaw J R 2010 BCIs in the laboratory and at home: the Wadsworth research program *Brain-Computer Interfaces* (Berlin: Springer) pp 97–111
- [15] www.intendix.com
- [16] Heinrich S and Bach M 2008 Signal and noise in P300 recordings to visual stimuli *Doc. Ophthalmol.* **117** 73–83
- [17] Mak J N et al 2011 Optimizing the P300-based brain-computer interface: current status, limitations and future directions *J. Neural Eng.* **8** 025003
- [18] Ortner R et al (ed) 2011 Accuracy of a P300 speller for people with motor impairments *2011 IEEE Symp. on Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB)* pp 1–6
- [19] Spüler M, Bensch M, Kleih S, Rosenstiel W, Bogdan M and Kübler A 2012 Online use of error-related potentials in healthy users and people with severe motor impairment increases performance of a P300-BCI *Clin. Neurophysiol.* **123** 1328–37
- [20] Riccio A et al 2013 Attention and P300-based BCI performance in people with amyotrophic lateral sclerosis *Frontiers Hum. Neurosci.* **2013** 7
- [21] Schreuder M, Hohne J, Treder M, Blankertz B and Tangermann M (ed) 2011 Performance optimization of ERP-based BCIs using dynamic stopping *2011 Annual Int. Conf. of the IEEE on Engineering in Medicine and Biology Society (EMBC)* pp 4580–3
- [22] Schreuder M, Hohne J, Blankertz B, Haufe S, Dickhaus T and Tangermann M 2013 Optimizing event-related potential based brain-computer interfaces: a systematic evaluation of dynamic stopping methods *J. Neural Eng.* **10** 036025
- [23] Serby H, Yom-Tov E and Inbar G F 2005 An improved P300-based brain-computer interface *IEEE Trans. Neural Syst. Rehabil. Eng.* **13** 89–98
- [24] Lenhardt A, Kaper M and Ritter H J 2008 An Adaptive P300-based online brain-computer interface *IEEE Trans. Neural Syst. Rehabil. Eng.* **16** 121–30
- [25] Liu T, Goldberg L, Gao S and Hong B 2010 An online brain-computer interface using non-flashing visual evoked potentials *J. Neural Eng.* **7** 036003
- [26] Thomas E, Clerc M, Carpentier A, Daucea E, Devlaminck D and Munos R (ed) 2013 Optimizing P300-speller sequences by RIP-ping groups apart *2013 6th Int. IEEE/EMBS Conf. on Neural Engineering (NER)* pp 1062–5
- [27] Haihong Z, Cuntai G and Chuanchu W 2007 Towards asynchronous brain-computer interfaces: a P300-based computational approach with statistical models *IEEE Trans. Biomed. Eng.* **55** 5067–70
- [28] Park J, Kim K and Jo S (ed) 2010 A POMDP approach to P300-based brain-computer interfaces *Proc. of the 15th Int. Conf. on Intelligent User Interfaces (ACM)* pp 1–10
- [29] Speier W, Arnold C, Lu J, Deshpande A and Pouratian N 2014 Integrating language information with a hidden Markov model to improve communication rate in the P300 speller *IEEE Trans. Neural Syst. Rehabil. Eng.* **22** 678–84
- [30] Orhan U, Hild K E, Erdogmus D, Roark B, Oken B and Fried-Oken M (ed) 2012 RSVP keyboard: an EEG based typing interface *2012 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)* doi:10.1109/ICASSP.2012.6287966
- [31] Jin J et al 2011 An adaptive P300-based control system *J. Neural Eng.* **8** 036006
- [32] Hohne J, Schreuder M, Blankertz B and Tangermann M (ed) 2010 Two-dimensional auditory p300 speller with predictive text system *Annual Int. Conf. of the IEEE on Engineering in Medicine and Biology Society (EMBC)* (31 August–4 September) pp 4185–8
- [33] Kübler A, Holz E and Kaufmann T 2013 Bringing BCI controlled devices to end-users: a user centred approach and evaluation ed J L Pons, D Torricelli and M Pajaro *Converging Clinical and Engineering Research on Neurorehabilitation* vol 1 (Berlin: Springer) pp 1271–4
- [34] Throckmorton C S, Colwell K A, Ryan D B, Sellers E W and Collins L M 2013 Bayesian approach to dynamically controlling data collection in P300 spellers *IEEE Trans. Neural Syst. Rehabil. Eng.* **21** 508–17
- [35] Mainsah B, Colwell K A, Collins L M and Throckmorton C S 2014 Utilizing a language model to improve online dynamic data collection in P300 spellers *IEEE Trans. Neural Syst. Rehabil. Eng.* **22** 837–46
- [36] Ryan D, Colwell K, Throckmorton C, Collins L and Sellers E 2013 Enhancing brain-computer interface performance in an ALS population: checkerboard and color paradigms *5th Int. BCI Meeting (Asilomar, CA)* doi:10.3217/978-3-85125-260-6-26
- [37] Balota D A et al 2007 The English lexicon project *Behav. Res. Meth.* **39** 445–59
- [38] Schalk G, McFarland D J, Hinterberger T, Birbaumer N and Wolpaw J R 2001 BCI2000: development of a general purpose brain-computer interface (BCI) system *IEEE Trans. Biomed. Eng.* **51** 1034–43
- [39] The CMU Pronouncing Dictionary 2013 (www.speech.cs.cmu.edu/cgi-bin/cmudict)
- [40] McFarland D J, Sarnacki W A and Wolpaw J 2003 Brain-computer interface (BCI) operation: optimizing information transfer rates *Biol. Psychol.* **63** 237–51
- [41] Blankertz B, Krauledat M, Dornhege G, Williamson J, Murray-Smith R and Müller K-R A 2007 Note on brain actuated spelling with the Berlin brain-computer interface *Universal Access in Human-Computer Interaction Ambient Interaction (Lecture Notes in Computer Science* vol 4555) ed C Stephanidis (Berlin: Springer) pp 759–68

- [42] Ryan D B, Frye G E, Townsend G, Berry D R, Mesa-G S, Gates N A and Sellers E W 2011 Predictive spelling with a P300-based brain–computer interface: increasing the rate of communication *Int. J. Hum.–Comput. Interact.* **27** 69–84
- [43] Kaufmann T, Völker S, Gunesch L and Kübler A 2012 Spelling is just a click away—a user-centered brain–computer interface including auto-calibration and predictive text entry *Frontiers Neurosci.* **6** 72
- [44] Nijboer F, Birbaumer N and Kubler A 2010 The influence of psychological state and motivation on brain–computer interface performance in patients with amyotrophic lateral sclerosis—a longitudinal study *Frontiers Neurosci.* **4**
- [45] Kleih S C, Nijboer F, Halder S and Kubler A 2010 Motivation modulates the P300 amplitude during brain–computer interface use *Clin. Neurophysiol.* **121** 1023–31
- [46] Lin E and Polich J 1999 P300 habituation patterns: individual differences from ultradian rhythms *Perceptual Motor Skills.* **88** 1111–25
- [47] Cedarbaum J M *et al* 1999 The ALSFRS-R: a revised ALS functional rating scale that incorporates assessments of respiratory function *J. Neurological Sci.* **169** 13–21
- [48] McCane L M *et al* 2014 Brain–computer interface (BCI) evaluation in people with amyotrophic lateral sclerosis *Amyotrophic Lateral Sclerosis Frontotemporal Degeneration* **15** 207–15
- [49] Treder M S and Blankertz B 2010 (C)overt attention and visual speller design in an ERP-based brain–computer interface *Behav. Brain Functions* **6** 28
- [50] Allison B and Neuper C 2010 Could anyone use a BCI? *Brain–Computer Interfaces (Human–Computer Interaction Series)* ed D S Tan and A Nijholt (London: Springer) pp 35–54
- [51] Sellers E W, Ryan D B and Hauser C K 2014 Noninvasive brain–computer interface enables communication after brainstem stroke *Sci. Translational Med.* **6** 257re7
- [52] Nijboer F *et al* 2008 A P300-based brain–computer interface for people with amyotrophic lateral sclerosis *Clin. Neurophysiol.* **119** 1909–16
- [53] Mainsah B, Morton K, Collins L and Throckmorton C (ed) 2014 Extending language modeling to improve dynamic data collection in ERP-based spellers *6th Int. Brain–Computer Interface Conf.* (Asimolar, CA) doi:10.3217/978-3-85125-378-8-80