Automatic motor task selection via a bandit algorithm for a brain-controlled button

2013 J. Neural Eng. 10 016012

(http://iopscience.iop.org/1741-2552/10/1/016012)

J. Neural Eng. **10** (2013) 016012 (8pp) doi:10.1088/1741-2560/10/1/016012

Automatic motor task selection via a bandit algorithm for a brain-controlled button

Joan Fruitet[1], Alexandra Carpentier[2], Rémi Munos[2] and Maureen Clerc[1]

[1] Athena Project Team, INRIA, Sophia Antipolis, France
[2] SequeL Project Team, INRIA, Lille, France

E-mail: Maureen.Clerc@inria.fr

Received 27 July 2012
Accepted for publication 9 November 2012
Published 21 January 2013
Online at stacks.iop.org/JNE/10/016012

Abstract
Objective. Brain–computer interfaces (BCIs) based on sensorimotor rhythms use a variety of motor tasks, such as imagining moving the right or left hand, the feet or the tongue. Finding the tasks that yield best performance, specifically to each user, is a time-consuming preliminary phase to a BCI experiment. This study presents a new adaptive procedure to automatically select (online) the most promising motor task for an asynchronous brain-controlled button. *Approach*. We develop for this purpose an adaptive algorithm UCB-classif based on the stochastic bandit theory and design an EEG experiment to test our method. We compare (offline) the adaptive algorithm to a naïve selection strategy which uses uniformly distributed samples from each task. We also run the adaptive algorithm online to fully validate the approach. *Main results*. By not wasting time on inefficient tasks, and focusing on the most promising ones, this algorithm results in a faster task selection and a more efficient use of the BCI training session. More precisely, the offline analysis reveals that the use of this algorithm can reduce the time needed to select the most appropriate task by almost half without loss in precision, or alternatively, allow us to investigate twice the number of tasks within a similar time span. Online tests confirm that the method leads to an optimal task selection. *Significance*. This study is the first one to optimize the task selection phase by an adaptive procedure. By increasing the number of tasks that can be tested in a given time span, the proposed method could contribute to reducing 'BCI illiteracy'.

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

1. Introduction

Scalp recorded electroencephalography (EEG) can be used for non-muscular control and communication systems, commonly called brain–computer interfaces (BCI). BCI systems based on sensorimotor rhythms (SMR) rely on the users' ability to control their SMR in the μ-frequency (8–13 Hz) and/or β-frequency (16–24 Hz) bands [1–3]. Indeed, these rhythms are naturally modulated during real and imagined motor action.

More precisely, real and imagined movements similarly activate neural structures located in the sensorimotor cortex, which can be detected in EEG recordings as synchronization (event-related synchronization (ERS)) and/or desynchronization (event-related desynchronization (ERD)) in the μ- and β-frequency bands [4, 5]. Because of the homuncular organization of the sensorimotor cortex [6], it is possible to distinguish different limb movements according to the position of the neural structures involved. For example, a right hand movement involves a modification of cortical activity on the upper left precentral gyrus, while the modification of activity due to feet movement is less lateralized.

A vast variety of motor tasks can be used in this context, like imagining rapidly or slowly moving the hand, grasping an object or kicking an imaginary ball. These different tasks permit different levels of control. Unfortunately, the tasks which enable optimal control of a BCI are specific to each user [7].

It is prevalent to use a preliminary training session to determine which motor task, among a small set of tasks, is the most appropriate for each user [8–10]. Although task selection is often mentioned as a preliminary step in the BCI literature, this paper is the first, to our knowledge, to propose a principled method for this purpose.

We now focus on a single brain-button controlled by imagining a specific motor task. The goal of the task selection is to find the motor task that will the most efficiently be detected with respect to the idle state.

The standard task selection method consists in requesting the subject to perform each motor task a fixed number of times (identical between tasks), and then to evaluate the performance of each task (its detection rate) in order to select the most appropriate. This naïve method is referred to in the rest of this paper as the *uniform strategy*.

We developed an adaptive algorithm that evaluates the performance of each task in real time during the experiment. It is therefore capable of rapidly eliminating non-efficient tasks in order to focus on the most promising ones. The benefit is twofold: reducing the length of the training stage and exploring a larger variety of motor tasks.

The task selection procedure is based on a bandit model (initially presented in [11]), which is assigned the goal to rapidly select an action that maximizes the expected detection rate given a limited budget of trials.

The rest of this paper is organized as follows. In section 2, we describe the EEG experiment and model the task selection as an optimization problem, which is solved using an Upper Confidence Bound (UCB) algorithm. Section 3 presents results on simulated and real online experiments, while section 4 discusses these results and presents perspectives. A proof of the theoretical performances of our algorithm is given in the appendix.

2. Material and methods

2.1. The EEG experiments

Two experiments were designed in order to evaluate the performance of our automatic method for task selection. The goal of the first experiment was to record a large amount of data to realize an offline performance analysis, whereas the second experiment was dedicated to testing the online use of the algorithm.

Both experiments were very similar and were designed to be as close as possible to the online use of a brain-controlled button. To this aim, we presented, at random timing, cue images during which the subjects had to perform 2 s long motor imagination tasks (intended to activate the button). During the online use of the adaptive algorithm, the order of the image presentations is optimized thanks to the adaptive strategy which is the subject of this paper, in view of selecting most rapidly the best motor task.

Ten subjects (aged 32.9 ± 9.4, all right-handed but one) participated in the offline experiment and four (all right-handed but one) underwent the online experiment. They had

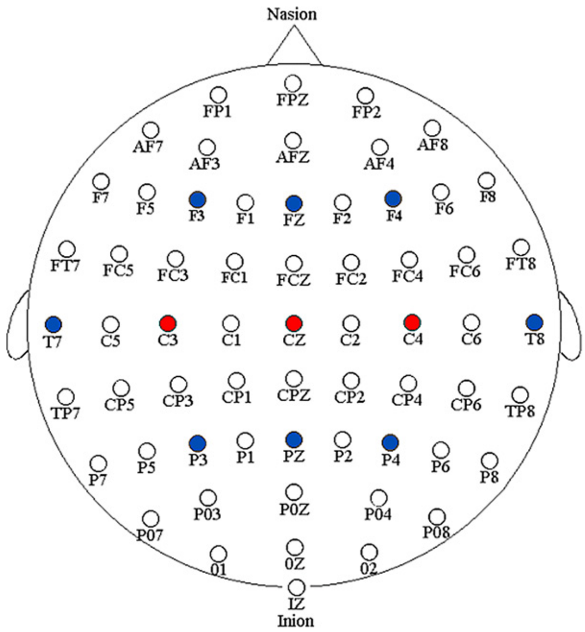

Figure 1. The 64 EEG cap with the three electrodes from which the features are extracted (C3, CZ, and C4). The additional electrodes used for the Laplacian are F3, FZ, F4, T7, T8, P3, PZ and P4.

no disability and were seated 1.5 m away from a 23′ LCD screen. Scalp electrodes were recorded through an OpenViBE platform [12] at a sampling rate of 512 Hz, from 11 out of 64 channels of a TMSI amplifier (see figure 1). The signal was band-pass filtered, and a spatial Laplacian was applied to increase the signal to noise ratio. EMG was recorded from both forearms and from the left leg to verify that the subjects did not activate their muscles during movement imagination.

The offline experiment was composed of 9–13 blocks of approximately 4 min. During each block, three cue images were presented ten times for 2 s in random order. The time between two image presentations varied between 2.5 and 9.5 s. The inter-trial interval was intentionally long, because it was used to acquire data in the no-control (idle) state, against which the tasks were to be classified. Each cue image was a prompt for the subject to imagine the corresponding motor action during 2 s, namely moving the right or the left hand or both feet (see figure 2).

During the online sessions, the *UCB-classif* algorithm and/or the *uniform strategy* were run several times to select one task among three (the right hand, both feet and the tongue). A budget of $N = 60$ task executions was dedicated to each task selection run. The rationale for using a limited number of tasks and a limited budget was to allow the task selection procedure to be run several times during one session. The goal was to verify that the *UCB-classif* algorithm could be used online and resulted in the selection of tasks that could be efficiently detected with respect to the idle state.

2.2. Feature extraction

In the case of short motor tasks, the movement imagination produces an ERD in the μ and β bands during the task,

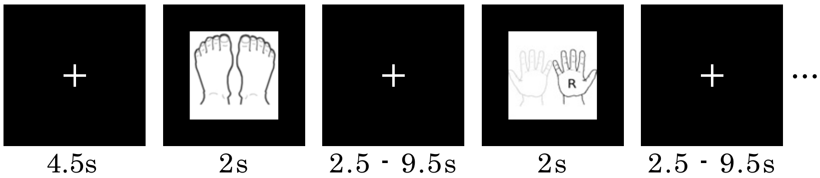

Figure 2. Time-line of the online and offline experiments. Cues are presented to the subjects for 2 s, with inter-cue intervals randomly varying from 2.5 to 9.5 s.

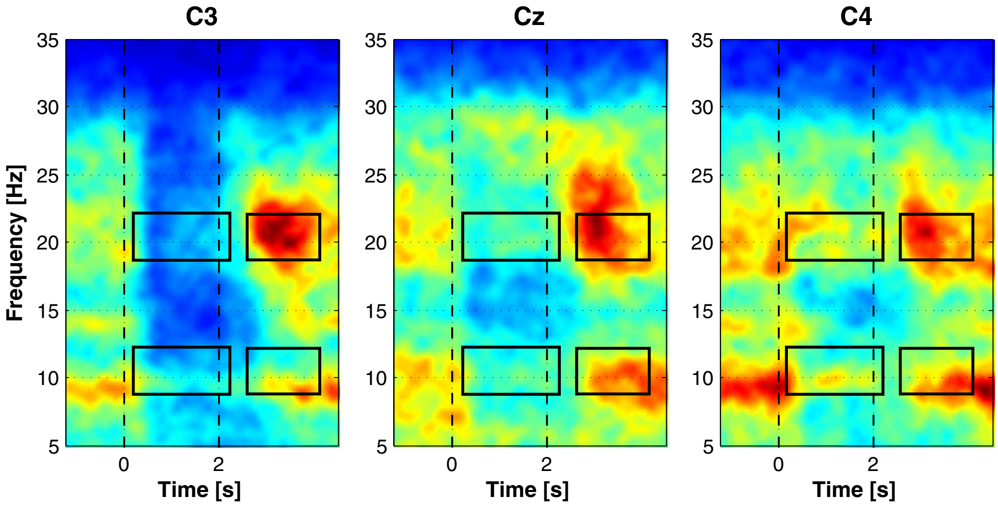

Figure 3. Time–frequency maps of the signal recorded on electrodes C3, Cz and C4, after spatial Laplacian, for a right hand movement imagination (presentation of the cue from $t = 0$ to $t = 2$ s). The maps have individual color maps, with maximal values 5.54, 3.28 and 4.69 on C3, Cz and C4, respectively (in arbitrary units). The time–frequency windows used for feature extraction are superimposed on the maps. During the movement, the power in the μ and β bands decreases (ERD) and, approximately 1 s after the movement, it increases (ERS) to reach a higher level than in the idle state.

followed by a strong ERS [13, 14] (sometimes called beta rebound as it is most usually seen in the β-frequency band).

We extracted features of the μ and β bands during the 2 s time windows of the motor action and on a subsequent 1.5 s of signal in order to use the bursts of μ and β power (ERS or rebound) that follow the movement imagination. More precisely, the features, indicated as rectangles in figure 3, were the power around 11 and 20 Hz (with 2 Hz wide bands) extracted at three electrodes over the sensorimotor cortex (C3, C4 and Cz) during and after the imagination. Thus, six features were extracted during the movement and six during the rebound (the benefit of using post-imagination features is explored in [14]). The time–frequency windows were chosen according to a preliminary study with the first subject and deliberately left unchanged for the other subjects.

Visual inspection of the signals showed no systematic eye-blinking or head-muscle clenching related to the cue presentations. We did not preprocess the data for such artifacts. This is also justified because the features used are specifically sensitive to motor activity of the limbs.

2.3. Evaluation of performances

The classifier was a linear SVM. The theoretical classification rate r_k^* of a task k is the probability that a new sample of data from this task would be well classified versus the idle condition, if an unlimited amount of data were available for training a linear SVM. Unfortunately, only a small amount of data is available to evaluate the performance of each task. To make the performance evaluation as precise as possible, we used the leave-one-out technique when less than 8 samples of the task were available, and an eightfold validation when more repetitions of the task had been recorded. More precisely, because of the large variance of the classification rate on the testing set when very little data are available, we computed the minimum between the classification rate over the training set (Train_i) and the testing set (Test_i), and then, evaluated its mean over the different folds:

$$\frac{1}{N_{\mathrm{folds}}} \sum_{i=1}^{N_{\mathrm{folds}}} \min(\mathrm{Train}_i, \mathrm{Test}_i).$$

Although this estimator is biased, it gives more reliable results when little data are available.

2.4. Artificial tasks

One of the goals of our algorithm is to be able to select the best task among a large number of tasks. However, for the offline experiment, only a limited number of tasks could be used (three), because we limited the length of the sessions in order not to tire the subjects, but still needed a sufficient number of samples for each task in order to perform a reliable offline analysis.

To demonstrate the usefulness of our method for a larger number of tasks, we decided to create artificial (degraded) tasks by mixing the features of one of the real tasks (the feet) with different proportions of the features extracted during the idle periods.

2.5. Modeling the problem

Let K denote the number of different images to be presented to the subject during the learning stage (K is thus the number of imaginary tasks) and N is the total number of images (the budget). Our goal is to find a selection strategy (i.e. that chooses at each time step $t \in \{1, \dots, N\}$ an image $k_t \in \{1, \dots, K\}$ to present) which allows us to select *in fine* the most discriminative task (i.e. with the highest classification rate in generalization). Note that, in order to learn an efficient classifier, we need as much training data as possible, so our selection strategy should rapidly select the most promising images in order to obtain more samples from these rather than from the others.

This issue is relatively close to the *stochastic bandit problem* [11, 15]. The classical *stochastic bandit problem* is defined by a set of K actions (pulling different arms of bandit machines). With each action a reward distribution is associated, which is initially unknown from the learner. At time $t \in \{1, \dots, N\}$, if we choose an action $k_t \in \{1, \dots, K\}$, we receive a reward sample drawn independently from the distribution of the corresponding action k_t. The goal of the stochastic bandit algorithm is to find a selection strategy which maximizes the sum of obtained rewards.

We model the K different images as the K possible actions (or arms), and we define the reward as the detection rate of the corresponding imaginary motor task, against the idle state. In the *bandit problem*, pulling a bandit arm directly gives a stochastic reward which is used to estimate the distribution of this arm. In our case, when we display a new image, we obtain a new data sample for the selected motor task, which provides an additional data sample to train or test the corresponding classifier and thus obtain a more accurate performance estimation. The main difference is that for the *stochastic bandit problem*, the goal is to maximize the sum of obtained rewards, whereas the present goal is to maximize the performance of the classifier. However, the strategies are similar: since the distributions are initially unknown, one should first explore all the tasks (exploration phase) but then rapidly select the best one (exploitation phase). This is called the *exploration–exploitation trade-off*. The next paragraph presents an algorithm to optimize this trade-off.

2.6. The UCB-classif algorithm

The image selection strategy is designed by using a variant of the UCB algorithm [15], which builds a high probability UCB on the reward value of each task, and selects at each time step the action corresponding to the reward with highest bound.

The upper bound $B_{k,t}$ (of action k at time t) is defined in the *stochastic bandit problem* as the sum of the empirical reward $\hat{r}_{k,t}$ and a confidence term which depends on the number of times $T_{k,t}$ action k has been chosen up to time t:

$$B_{k,t} = \hat{r}_{k,t} + \sqrt{\frac{a \log N}{T_{k,t-1}}}, \qquad (1)$$

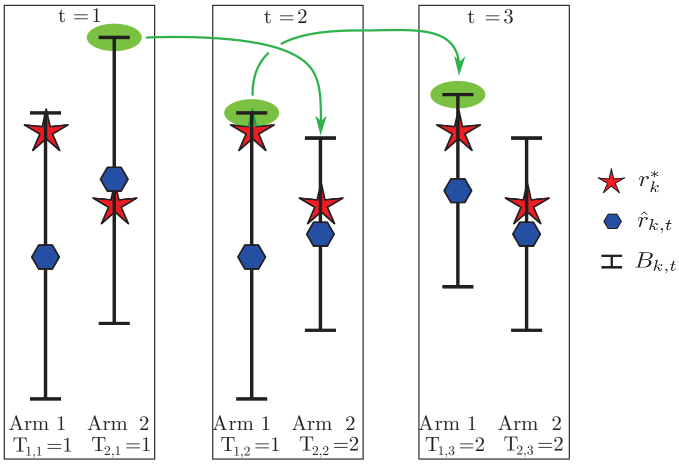

Figure 4. This figure represents three snapshots, at times $t = 1, 2$ and 3, of a bandit with two arms. Although arm 1 is the best arm ($r_1^* > r_2^*$, represented by the red stars), at time $t = 1$, since $B_{1,t} < B_{2,t}$, arm 2 is selected. Pulling arm 2 gives a better estimate $\hat{r}_{2,2}$ of r_2^* and reduces the confidence interval. At times $t = 2$ and $t = 3$, $B_{1,t}$ will be greater than $B_{2,t}$, so arm 1 will be selected.

Table 1. Pseudo-code of the *UCB-classif* algorithm.

The ***UCB-Classif*** **Algorithm**
Parameters: a, N, q
Present each image q times (thus set $T_{k,qK} = q$).
for $t = qK + 1, \dots, N$ **do**
  Evaluate by a q-split cross-validation the performance $\hat{r}_{k,t}$ of each image.
  Compute the UCB: $B_{k,t} = \hat{r}_{k,t} + \sqrt{\frac{a \log N}{T_{k,t-1}}}$ for each image $1 \leqslant k \leqslant K$.
  Present image: $k_t = \arg\max_{k \in \{1,\dots,K\}} B_{k,t}$.
  Update T: $T_{k_t,t} = T_{k_t,t-1} + 1$ and $\forall\, k \neq k_t$, $T_{k,t} = T_{k,t-1}$
end for

where $a > 0$ is a constant. The upper bound in formula (1) represents a compromise between the empirical reward (first term) and its uncertainty, which decreases with time (second term) (see figure 4 for an illustration).

We adapt the idea of the UCB to the adaptive classification problem and call this algorithm *UCB-classif* (see the pseudo-code in table 1). The algorithm builds the $B_{k,t}$-values (1), where $\hat{r}_{k,t}$ represents an estimation of the classification rate, by a q-fold cross-validation. The cross-validation uses a linear SVM classifier based on the $T_{k,t}$ data samples obtained (at time t) from movement k. Writing r_k^* the classification rate for the optimal linear SVM classifier (which would be obtained by using a infinite number of samples), we have the property that $B_{k,t}$ is a high probability upper bound on r_k^*: the probability $p(B_{k,t} < r_k^*)$ decreases to zero polynomially fast with N (see the proof in appendix A.1). The constant a is a measure of complexity (VC-dimension) of the class of linear SVM classifiers.

The choice of the initial number of task presentations q should be made with caution. If q is very small ($q < 5$), the algorithm may not give a fair chance to all the tasks. This can result in the elimination of the best task (although statistically it does not happen often). Selecting a value of q between 8 and 10 circumvents this problem.

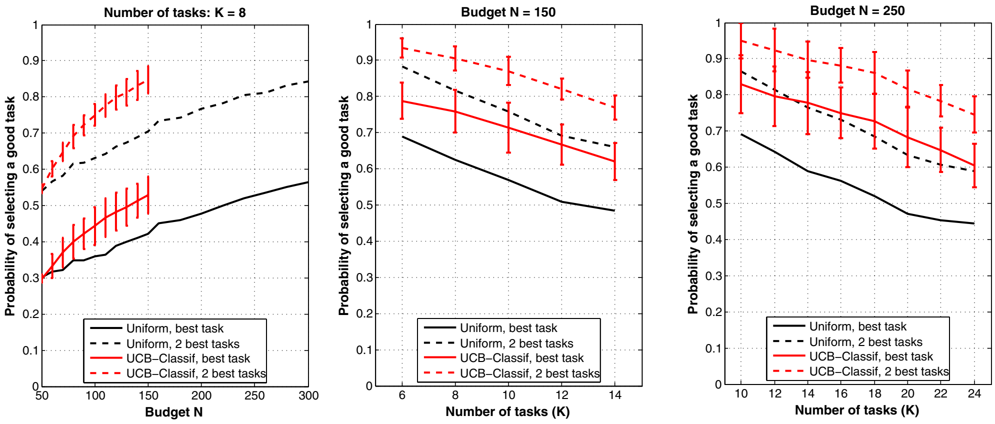

Figure 5. Probability of selecting the best task (full line), or one of the two best tasks (dashed line), as a function of the budget N (left graph) or of the number of tasks K (center and right graphs), on average over all ten subjects. The *UCB-classif* algorithm (in red) enables a more reliable selection than the *uniform strategy* (in black). The confidence interval with $p < 0.05$ of the gain provided by *UCB-classif* is represented as error bars.

Figure 4 is a graphical illustration showing how the *UCB-classif* algorithm works. The idea behind the algorithm is that it selects at time t an image k either because it has a good classification rate $\hat{r}_{k,t}$ (thus it is interesting to obtain more samples from it to perform exploitation) or because it has not been sampled many times, making its classification rate uncertain (thus it is important to explore it more). In fine, this allows the image that has the highest classification rate to be sampled more often. The *UCB-classif* algorithm guarantees that non-optimal images are presented only a negligible fraction of times ($\log N$ times out of a total budget N). The best image (or motor task) is thus sampled $N - O(\log N)$ times (see the proof in appendix A.2).

3. Results

3.1. Offline performance analysis of UCB-classif

We compare the performance of the *UCB-classif* sampling strategy to the *Uniform strategy*, i.e. the standard way of selecting a task, consisting of N/K presentations of each image and, once all the data have been acquired, selecting the task that maximizes the classification accuracy. To obtain a precise evaluation of the performance, we use a bootstrap technique: for each budget N and number of tasks K, we simulate 500 online selections by randomly choosing the data used by the *UCB-classif* algorithm or the *uniform strategy*. We then compute the probability of selecting the best-performing tasks and compare them between methods. The best-performing tasks were defined by ranking the task performances based on the whole dataset, and even on other recording sessions when they were available (which was the case for seven of the ten offline subjects). Because for some subjects, the two best-ranked tasks have very close performance, we computed both the probability of selecting the best task and the probability of selecting one of the best two tasks.

Figure 5 shows these probabilities according to different budgets and numbers of tasks. The *UCB-classif* strategy significantly outperforms the *uniform strategy*, even for relatively small N.

When eight tasks are used (left graph of figure 5), the probability of selecting the best task (and one of the two best tasks) with *UCB-classif* and a budget of $N = 150$ is similar or higher to those obtained by the *uniform strategy* for a budget of $N = 250$. Thus, only 150 trials are needed to select a good task instead of 250. For this application, the time spent to select the task is reduced from 35 to 20 min with no loss of reliability.

For a budget of $N = 250$ (right graph of figure 5), the number of tasks that can be explored with the same reliability rises from 10 to around 20 when the *UCB-classif* algorithm is used instead of the *uniform strategy*.

One can note that our algorithm does not guarantee that the task selected is the best performing, indeed the probabilities shown in figure 5 are not all equal to one. But the chance of selecting a non-optimal task is lower for *UCB-classif* than for the *uniform strategy* and becomes lower as N/K becomes large. A strategy to avoid selecting a non-optimal task is to increase the budget N, which represents a compromise between spending time on task selection and risking to select a task with non-optimal performance.

3.2. Online performance analysis of UCB-classif

In order to accurately measure the gain in using *UCB-classif* compared to the *uniform strategy*, at least 100 task selections should be made with both techniques on a large group of subjects. This would require hundreds of hours of recordings for each subject, which is not feasible. This is why we have used an offline analysis to precisely evaluate the performance of *UCB-classif*. We nevertheless designed an online experiment for the purpose of verifying that the *UCB-classif* algorithm could indeed be used online and that it leads to the selection of well-performing tasks. For this online evaluation, we repeated between 3 and 5 times, for each subject, the selection of one task among the three with *UCB-classif* [3].

[3] For the first subject, one task selection was also made using the *uniform strategy*, but since the purpose of the online experiment was not to evaluate the *uniform strategy*, this was not reproduced with the other subjects.

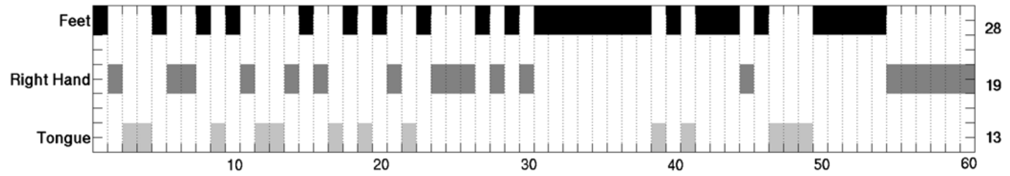

Figure 6. Example of a task sequence delivered during one run of the online session (recorded on subject *A4*, last run reported in table 2).

Table 2. Tasks selected during online experiments for four subjects. A budget of $N = 60$ was used to determine the best task among $K = 3$. The number of presentations Nb and the final detection rate of each task is indicated for each experiment.

Subject	Right hand		Feet		Tongue		Selected
*A*1	Nb	Rate	Nb	Rate	Nb	Rate	task
Uniform	20	79%	20	75%	20	57%	Right hand
UCB-classif	23	77%	27	79%	10	48%	Feet
UCB-classif	16	70%	34	88%	10	74%	Feet
UCB-classif	21	75%	30	82%	9	61%	Feet
*A*2							
UCB-classif	26	88%	20	86%	14	78%	Right hand
UCB-classif	25	78%	23	76%	12	69%	Right hand
UCB-classif	22	63%	26	70%	12	60%	Feet
UCB-classif	27	89%	22	84%	11	71%	Right hand
UCB-classif	23	80%	24	83%	13	57%	Feet
*A*3							
UCB-classif	24	79%	15	56%	21	64%	Right hand
UCB-classif	42	94%	11	67%	7	63%	Right hand
UCB-classif	35	87%	14	69%	11	54%	Right hand
UCB-classif	26	84%	21	61%	13	50%	Right hand
UCB-classif	32	78%	17	74%	11	43%	Right hand
*A*4							
UCB-classif	16	80%	28	97%	16	78%	Feet
UCB-classif	19	72%	30	87%	11	40%	Feet
UCB-classif	16	70%	34	86%	10	57%	Feet
UCB-classif	15	64%	34	88%	11	55%	Feet
UCB-classif	19	73%	28	86%	13	76%	Feet

The observations made during the online recordings can be found in table 2, namely for each run: the number of presentations of each task, final detection rate of each task and the task selected. Figure 6 gives an example of task sequence delivered during one of the runs.

For three out of four subjects, each experiment with the *UCB-classif* algorithm led to the selection of the best task, whereas the unique attempt of using the *uniform strategy* resulted in the selection of a non-optimal task. For subject *A*2, the performances of the right hand and the feet were very similar, which makes it irrelevant to determine the 'best' task. The *UCB-classif* algorithm alternatively selected one or the other, and always succeeded in rapidly eliminating the worst task (the tongue). Observe that when tasks have similar performance, they end up having similar presentation rates.

For all subjects, the performance of the tongue is lower than the two other tasks, and we can observe that the *UCB-classif* algorithm allocates only a small budget to the tongue and focuses on evaluating the detection rate of the right hand and the feet.

In addition to the experiments reported in table 2, subject *A*3 performed an extra run, with five instead of three tasks (right hand, left hand, feet, tongue, right arm movement) and a budget of $N = 120$. It resulted in the selection of the best task, namely the right hand.

4. Discussion and perspectives

The results of the *UCB-classif* algorithm on the offline analysis and the online sessions are very promising. Because it is not penalized by non-efficient tasks, *UCB-classif* allows the exploration of larger sets of motor tasks than customary up to now. For example, for eight tasks, on average only 150 trials are necessary, whereas 250 would be needed with the naïve, uniform, task selection. This can automatically reduce the time required for this selection phase, with no loss of reliability. In order to fully validate the method, we have applied it online[4]. The tasks selected online achieved detection rates between 70% and 97%, with an average of 84.6%.

The use of *UCB-classif* is all the more worthwhile as the number of tasks is large: for only three or four tasks, as customary in BCI today, it offers little advantage over uniform task selection. But this new selection method, by accommodating a large number of tasks, allows us to explore different movement strategies for a given limb (e.g. kicking, rotating, swaying, repeating brisk movements).

For this study, we have chosen to use a very small set of fixed features (12 features, extracted from three electrodes, two frequency bands and two time windows) that were calibrated on only one subject during a preliminary experiment. The detection rate could be further improved by using a larger set of subject-specific features [16] and more advanced techniques (like the CSP [17] or feature selection [18]). This is why, once the best task has been determined using *UCB-classif*, the acquired data should be used to automatically adjust the features for each subject. By rapidly focusing, during this exploratory phase, on the most discriminative task, sufficient data will be available to appropriately tune the features, in addition to training the classifier.

An alternative strategy would be to start tuning the features while running *UCB-classif*. Unfortunately, this would give rise to two issues: first, an important risk of over-fitting, especially for an initially very small amount of data, second a risk of favoring the tasks that have been the most sampled, and for which the features will thus be the best tuned.

The BCI targeted in this paper uses only one motor task to control a single brain-button. It would be interesting to aim for BCI that use two or more tasks to control more buttons. With this in view, we are studying an extended version of *UCB-classif* that will be able to select a pair (or a triplet) of tasks

[4] The code, which runs with the OpenViBE software platform, can be made available upon request.

that will maximize both the detection rate of each task against idle state and the classification rate between the two (or three) tasks.

We have developed a variant of the algorithm for which the budget N does not need to be specified [19]. The algorithm automatically finds the optimal number of presentations in order to select the best task with a fixed probability of confidence. Unfortunately, the differences between classification accuracies are generally too small, and this automatic adjustment of the budget results in long experiments. We think that a good compromise is to start with a fixed budget, and for a human observer to decide when to stop the selection, based on the classification performance as well as on the level of commitment and the tiredness of the subject.

Acknowledgments

This work was partially supported by the French National Research Agency grant Co-Adapt ANR-09-EMER-002. We would like to thank the reviewers for their constructive remarks.

Appendix. The *UCB-classif* algorithm

A proof of the upper bound used in *UCB-classif* is given in appendix A.1. This result is then used in appendix A.2 to calculate the theoretical performances of *UCB-classif.*

A.1. High probability bound for the classification error

Let $\mathcal{D}$ be a probability distribution in $\mathbb{R}^d \times \{0, 1\}$. Let $\mathcal{H}$ be the set of binary linear classifiers in $\mathbb{R}^d$, i.e. if $(x, y) \sim \mathcal{D}$, (i.e. are drawn from $\mathcal{D}$); then, $h(x)$ is the inferred class of the sample while the true class is y.

We define the $\{0, 1\}$ loss of a classifier h as

$$L_{\mathcal{D}}(h) = \mathbf{E}_{(x,y)\sim\mathcal{D}}(\mathbf{1}_{h(x)\neq y}).$$

Let h^* be the best linear classifier on $\mathcal{D}$ for the $\{0, 1\}$ loss:

$$h^* = \arg\min_{h\in H} L_{\mathcal{D}}(h).$$

Let now $\mathcal{X} = \{(x_1, y_1), \ldots, (x_T, y_T)\}$ be T i.i.d. points in $\mathbb{R}^d \times \{0, 1\}$, sampled from $\mathcal{D}$. We define the $\{0, 1\}$ empirical loss of a classifier h as

$$\hat{L}_{\mathcal{X}}(h) = \frac{1}{T}\sum_{t=1}^{N}(1_{h(x_t)\neq y_t}).$$

Let $\widehat{h^*} \in \mathcal{H}$ be the best empirical classifier on $\mathcal{X}$ in $\mathcal{H}$ for the empirical $\{0, 1\}$ loss:

$$\widehat{h^*} = \arg\min_{h\in\mathcal{H}} \hat{L}_{\mathcal{X}}(h).$$

Theorem 1 (Vapnik 1982)**.** *We have with the probability* $1-2\delta$ *that*

$$|L_{\mathcal{D}}(h^*) - \hat{L}_{\mathcal{X}}(\widehat{h^*})| \leqslant 2\sqrt{\frac{d(\log(2T/d)+1)+\log(4/\delta)}{T}}.$$

In our setting and for task k, $1 - r_k^*$ is the $\{0, 1\}$ loss of the best classifier for task k and $1 - \hat{r}_{k,t}$ is the empirical $\{0, 1\}$ loss of the empirical best classifier for task k with $T_{k,t}$ samples. We thus have with the probability $1 - 2\delta$ that for task k

$$|r_k^* - \hat{r}_{k,t}| \leqslant 2\sqrt{\frac{d(\log(2T_{k,t}/d)+1)+\log(4/\delta)}{T_{k,t}}}, \tag{A.1}$$

where d is the dimension of the feature space, which in this paper is 12. Let us now choose $\delta = 1/N^2$. Using (A.1), $T_{k,t} < N$, we have with the probability $1 - 2/N^2$ that

$$|r_k^*-\hat{r}_{k,t}| \leqslant 2\sqrt{\frac{d(\log(2N/d)+1)+\log(4N^2)}{T_{k,t}}} \leqslant \sqrt{\frac{a\log(N)}{T_{k,t}}},$$

where $a = 6(d + 1)$ when N is big enough. We thus have with the probability $1 - 2/N^2$ that

$$r_k^* \leqslant \hat{r}_{k,t} + \sqrt{\frac{a\log(N)}{T_{k,t}}} \leqslant r_k^* + 2\sqrt{\frac{a\log(N)}{T_{k,t}}}. \tag{A.2}$$

This proves that $B_{k,t} = \hat{r}_{k,t} + \sqrt{\frac{a\log(N)}{T_{k,t}}}$ is an upper bound with the high probability on r_k^*.

A.2. Theoretical performances of UCB-classif

In the event of large probability that $B_{k,t}$ is an upper bound on r_k^* for any k and any N large enough, we know that we pull at time t a sub-optimal arm k if, for the best arm k^* with reward r^*, $B_{k^*,t} \leqslant B_{k,t}$.

According to (A.2) this implies that

$$r^* \leqslant B_{k^*,t} \leqslant B_{k,t} \leqslant r_k^* + 2\sqrt{\frac{a\log(N)}{T_{k,t}}}.$$

This means by a simple computation that we pull a sub-optimal arm k only if

$$T_{k,t} \leqslant 4\frac{a\log(N)}{(r^* - r_k^*)^2}.$$

Since $T_{k,t}$ is the number of times arm k is pulled, we thus pull the suboptimal arms only a number of times in $O(\log(N))$ and, finally, pull the optimal arm $N - O(\log(N))$ times, more precisely at least $N-\sum_{k\neq *} 4\frac{a\log(N)}{(r^*-r_k^*)^2}$ times. Moreover, the error of the empirical classifier on the best arm is such that, with high probability,

$$|r^* - \hat{r}^*| \leqslant \sqrt{\frac{a\log(N)}{N - \sum_{k\neq *} 4\frac{a\log(N)}{(r^*-r_k^*)^2}}}.$$

References

[1] McFarland D J, Sarnacki W A and Wolpaw J R 2010 Electroencephalographic (EEG) control of three-dimensional movement *J. Neural Eng.* **7** 036007

[2] Solis-Escalante T, Müller-Putz G, Brunner C, Kaiser V and Pfurtscheller G 2010 Analysis of sensorimotor rhythms for the implementation of a brain switch for healthy subjects *Biomed. Signal Process. Control* **5** 15–20

[3] Blankertz B, Dornhege G, Krauledat M, Müller K-R and Curio G 2007 The non-invasive Berlin brain–computer interface: fast acquisition of effective performance in untrained subjects *NeuroImage* **37** 539–50

[4] Pfurtscheller G and da Silva F H L 1999 Event-related EEG/MEG synchronization and desynchronization: basic principles *Clin. Neurophysiol.* **110** 1842–57

[5] Pfurtscheller G and Neuper C 1997 Motor imagery activates primary sensorimotor area in humans *Neurosci. Lett.* **239** 65–8

[6] Jasper H and Penfield W 1949 Electrocorticograms in man: effect of voluntary movement upon the electrical activity of the precentral gyrus *Archiv für Psychiatrie und Nervenkrankheiten* **183** 163–74

[7] Dobrea M-C and Dobrea D M 2009 The selection of proper discriminative cognitive tasks—a necessary prerequisite in high-quality BCI applications *ISABEL 2009: Proc. 2nd Int. Symp. on Applied Sciences in Biomedical and Communication Technologies* pp 1–6

[8] Wolpaw J R, Birbaumer N, McFarland D J, Pfurtscheller G and Vaughan T M 2002 Brain–computer interfaces for communication and control *Clin. Neurophysiol.* **113** 767–91

[9] Millán J d R, Renkens F, Mourino J and Gerstner W 2004 Brain-actuated interaction *Artif. Intell.* **159** 241–59

[10] Vidaurre C and Blankertz B 2010 Towards a cure for BCI illiteracy *Brain Topogr.* **23** 194–8

[11] Robbins H 1952 Some aspects of the sequential design of experiments *Bull. Am. Math. Soc.* **58** 527–35

[12] Renard Y, Lotte F, Gibert G, Congedo M, Maby E, Delannoy V, Bertrand O and Lécuyer A 2010 OpenViBE: An open-source software platform to design, test, and use brain–computer interfaces in real and virtual environments *Presence: Teleoperators Virtual Environ.* **19** 35–53

[13] Fruitet J, Clerc M and Papadopoulo T 2011 Preliminary study for an hybrid BCI using sensorimotor rhythms and beta rebound *Int. J. Bioelectromagn.* **13** 70–1

[14] Thomas E M, Fruitet J and Clerc M 2012 Investigating brief motor imagery for an ERD/ERS based BCI *EMBC'12: Proc. 34th Int. Conf. IEEE Engineering in Medicine and Biology Society* pp 2929–32

[15] Auer P, Cesa-Bianchi N and Fischer P 2002 Finite time analysis of the multiarmed bandit problem *Mach. Learn.* **47** 235–56

[16] Dobrea M, Dobrea D M and Alexa D 2010 Spectral EEG features and tasks selection process: some considerations toward BCI applications *Proc. IEEE Int. Workshop on Multimedia Signal Processing* pp 150–5

[17] Ramoser H, Muller-Gerking J and Pfurtscheller G 2000 Optimal spatial filtering of single trial EEG during imagined hand movement *IEEE Trans. Rehabil. Eng.* **8** 441–6

[18] Fruitet J, McFarland D J and Wolpaw J R 2010 A comparison of regression techniques for a two-dimensional sensorimotor rhythm-based brain–computer interface *J. Neural Eng.* **7** 016003

[19] Fruitet J, Carpentier A, Munos R and Clerc M 2012 Bandit algorithms boost brain–computer interfaces for motor-task selection of a brain-controlled button *Advances in Neural Information Processing Systems (NIPS)* (Curran Associates, Inc.) to be published