Proceedings of the
10th Graz Brain-Computer Interface Conference 2026
10.3217/978-3-99161-093-9-005

Realtime Class Balancing for Efficient Decoder Adaptation in Asynchronous Brain-Computer Interface

Rémi Souriau[1], Félix Martel[1], Thomas Costecalde[1], Serpil Karakas[1], Tetiana Aksenova[1]

[1]Univ. Grenoble Alpes, CEA, Leti, Clinatec, Grenoble, France

E-mail: tetiana.aksenova@cea.fr

ABSTRACT: Brain-Computer Interfaces (BCIs) allow direct control of external effectors via brain activity decoding. For real-world use, BCIs must operate in an asynchronous mode that enables users to initiate or withhold system use, while maintaining low false positive rates (FPR) and high decoding accuracy. Class balance is crucial for efficient model training. The RSW-NPLS algorithm has been proposed recently to balance classes during online and incremental learning. However, perfect balancing may not be optimal for asynchronous BCIs, as they require a highly stable idle state. We propose a class balancing strategy that prioritizes the idle state to reduce its false activations of the other states, and evaluated it on four databases from tetraplegic and paraplegic users. It significantly reduced the FPR (between -55.36% and -70.08%) with minimal loss of accuracy (between -3.68% and -10.51%).

INTRODUCTION

A Brain-Computer Interface (BCI) enables a person to control an effector, such as a wheelchair, an exoskeleton [1] or an electrical stimulator [2], by performing motor imagery or attempting movements. This control is achieved by recording the person's neuronal activity. Brain signals are processed, and a decoder predicts the user's intention to control an effector. Motor BCI systems can be used to compensate lost functions or in rehabilitation programs [3].

Efficient decoding of user's motor intention remains a major challenge to guarantee satisfactory performance of BCI systems. Numerous decoding algorithms were proposed to address this challenge. However, many of them are limited to operating in a synchronous mode within controlled laboratory environments. In such experiments, the activation time of the BCI system is predetermined and fixed by the system. In contrast, autonomous BCIs suitable for everyday use should support an asynchronous mode, including an idle state (or resting state). Users should be able to activate and to deactivate the BCI system at their own will, and it is essential to prevent / minimize the unintended effectors activation. This makes the Idle State (IS) distinct from other Active States (AS). In addition, practical BCI systems should be compact (i.e., portable) and capable of online operation for both prediction and decoder training. The learning algorithm has to work online to take into consideration the user feedback, and has to support learning since new states can be added to the decoder after some update iterations.

Some tasks may be complex and require longer training times than other states. In addition, participant training often require rest periods to avoid user fatigue. For these reasons, the experimenter cannot properly balance the different states, leading to class imbalance in the training data and degraded decoding performance [4]. In the case of real-time, streaming learning of the BCI decoder, class balancing must be performed online. This poses a challenge because not all data are known, and therefore the final class imbalance is unknown. Despite of the importance of class balancing, this problem has been rarely addressed in BCI studies, *e.g.* [4–6]. In [4], the proposed Recursive Sample Weighted N-way Partial Least Squares regression (RSWNPLS) is integrated to a Hidden Markov Model (HMM). The HMM-RSWNPLS algorithm is well adapted to online updates supporting class balancing for multi-class classification during incremental learning, while being compact. It incorporates a weighting mechanism in order to balance each class during incremental learning of the decoding model, and significantly improves the classification accuracy compared to the same algorithm without balancing mechanism. However, HMM-RSWNPLS, and more generally algorithms to control class balance in BCIs [4–6], treat the IS like any other state. This solution may be suboptimal for asynchronous BCI where preventing false activation of the system is one of critical issue.

In this study, we propose a class balancing strategy to target any class ratio, while incrementally learning the decoding model. More specifically, we studied the impact of balancing towards the IS to promote a better stability and limit false activations of the BCI. The proposed strategy was integrated into the HMM-RSWNPLS and tested in a pseudo-online manner on 4 different databases recorded from one tetraplegic and two paraplegic participants. The unwanted activation of the BCI is quantified with the False Positive Rate (FPR). We evaluated variations of the FPR and the decoding accuracy in function of the balance between the IS and AS. Results show that the proposed algorithm can reduce significantly the FPR without significantly degrading the balanced accuracy. It also decreases the global latency and the noise level of IS predictions.

CC BY 4.0 - https://creativecommons.org/licenses/by/4.0/
This CC license does not apply to third-party material and content noted otherwise.

10.3217/978-3-99161-093-9-005

MATERIALS AND METHODS

Decoders for Asynchronous BCI:

Most BCI decoders consist in classification models where each class (or state) corresponds to a task that the user can perform. Numerous classifiers have been evaluated, including both static and dynamic approaches. Sample-wise static classifiers like linear discriminant analysis, support vector machine, logistic regression, k-nearest neighbors, or deep neural networks have been widely tested [7, 8]. Since asynchronous BCI decoders aim to decode temporal signal, dynamic classifiers are preferred [9] to capture temporal dependencies in recorded signals. In [10] tests show that taking the dynamic of data into account improves the decoder performance. HMM [1, 2] and Hierarchical HMM [11] have been proposed to decode user intents in an online manner. One of the main issues in asynchronous BCI systems is ensuring the accurate decoding of the IS to prevent false activations, where the system performs an action when the user does not intend it. Several studies have proposed BCI systems designed for asynchronous operation to control various effectors, such as robotic arms [12], wheelchairs [13], or environmental control systems [14]. These studies have explored diverse strategies for managing the IS. In [12], the authors used a threshold-based method to distinguish IS from AS on ElectroEncephaloGram (EEG) signals. Another approach uses a heuristic rule to detect the IS [13], and if the IS is not detected, a classifier is applied to find the desired AS. Another strategy uses hierarchical classifiers [11], a first classification model separates the IS from the AS, followed by a subsequent classification of the AS. The presented approaches aim to separate the IS from AS to minimize false activations. However, these methods do not allow to control the trade-off between decoding accuracy and false activations, and may inadvertently favor IS over AStoo much. As a result, the decoder could become more likely to overlook an AS, potentially leading to a decline in overall accuracy.

Studies cited bellow evaluate model accuracy and the number or ratio of false activations, but, to the best of our knowledge, none incorporate a mechanism in the decoder to allow users to control the trade-off between accuracy and false activation limitation. In this study, the HMM-RSWNPLS algorithm, proposed in [4] is adapted to control the ratio between the IS and the AS to limit false activations and maintain a good classification accuracy in asynchronous BCI.

HMM-RSWNPLS algorithm:

The HMM is a dynamic classifier adapted for multi-class classification tasks. It is composed of two stochastic processes: the latent process $\{Y_t\}_{t=1}^{T}$ and the observation $\{X_t\}_{t=1}^{T}$ process which refers to, respectively, the user intent and the neural features. Two probabilistic laws are learnt to model the dynamic evolution of these processes: the emission law that provides $\Pr(X_t|Y_t)$ and the transition law $\Pr(Y_t|Y_{t-1})$. The transition law is modelled using a transition matrix $A \in \mathbb{R}^{(K\times K)}$, where K is the number of states/classes of the model. Each coefficient of the matrix $a_{ij} = \Pr(Y_t = j|Y_{t-1} = i)$ is learnt by counting the transitions during the training session. The RSWNPLS [4] is used to learn the emission model. This algorithm learns a linear relationship between the user intent Y_t and the neural feature X_t in an incremental way. Data are acquired buffer by buffer and are discarded after each update of the model to keep the memory load stable.

The model allows to compute a probability for each state at every time index t, resulting in a prediction probability function over time for each state [1]. First, the output of the RSWNPLS is computed resulting in an estimation vector $\widehat{Y}_t$. The probability of the $k-$th state $\Pr(Y_t = k|X_t)$ is then deduced from $\widehat{Y}_t$ using the soft-max function. In this study, a scaling parameter θ is applied in soft-max function [15]: $\Pr(Y_t|X_t) = \text{soft-max}\left(\theta \times \widehat{Y}_t\right)$. The prediction probability $\Pr(Y_t = k|X_{1:t})$ (where $X_{1:t}$ is the sequence of observation from the initial time step to t) is computed using the Bayes' rule and the HMM's assumption (first-order Markov model).

To compensate potential class imbalance, the RSWNPLS weights samples during the learning of the model. The weighting procedure is designed such that weighting data is equivalent to resampling the data, but can use non-integer values, which allows for greater flexibility in strategies to balance classes. Let $N_u^{(k)}$ be the cumulated size of the class k at iteration update u, $n_u^{(k)}$ be the size of the class k in the current buffer at iteration update u, λ be the forgetting factor, and $w_u^{(k)}$ be the weight applied to each element in the class k of the current buffer at iteration update u. For each k, the update rules of $N_u^{(k)}$ is given by:

$$N_u^{(k)} = \lambda N_{u-1}^{(k)} + w_u^{(k)} n_u^{(k)}. \tag{1}$$

To balance classes in [4], the weight applied to the current majority class is equal to 1. Weights of other classes are computed relatively to the majority one:

$$w_u^{(k)} = \frac{N_u^{(maj)} - \lambda N_{u-1}^{(k)}}{n_u^{(k)}} \tag{2}$$

where $N_u^{(maj)}$ is the size of the current majority class:

$$N_u^{(maj)} = \max_k \left(\lambda N_{u-1}^{(k)} + n_u^{(k)}\right). \tag{3}$$

A saturation is applied to $w_u^{(k)}$ to limit the maximum value of weights to avoid a few samples getting excessively high weights.

Modification on the RSWNPLS to control the class imbalance ratio:

The weight computation presented in the previous section only allows to balance all classes without any distinction. The weighting computation is modified in this study to control the class imbalance during the model learning. A target ratio between classes is defined, and the weighting

CC BY 4.0 - https://creativecommons.org/licenses/by/4.0/
This CC license does not apply to third-party material and content noted otherwise.

10.3217/978-3-99161-093-9-005

computation aims to maintain the ratio of between classes during the incremental learning of the decoder.
Let $\{r^{(k)}\}_{k=1}^{K}$, subject to $\sum_{k=1}^{K} r^{(k)} = 1$ and $r^{(k)} > 0$ for all k, be the desired ratio of each of the K classes. The updated weight computation is given by:

$$w_u^{(k)} = \frac{r^{(k)} N_u^{(maj)} - \lambda N_{u-1}^{(k)}}{n_u^{(k)}} \tag{4}$$

where $N_u^{(maj)}$ is the size of the current majority class given by:

$$N_u^{(maj)} = \max_k \left(\frac{\lambda N_{u-1}^{(k)} + n_u^{(k)}}{r^{(k)}} \right). \tag{5}$$

By replacing $w_u^{(k)}$ in equation (1) with its expression from equation (4), one can show $N_u^{(k)} / \sum_j N_u^{(j)} = r^{(k)}$. This weight computation allows to control the class imbalance between states. Similarly to the generic algorithm [4], a saturation is imposed on the weights to bound their maximal value.

Database:
Four databases have been used to test the proposed strategy. The first two come from the “BCI and Tetraplegia” clinical trial (ClinicalTrials.gov, NCT02550522) [1], the third database come from the “STIMO-BSI” clinical trial (clinicaltrials.gov, NCT04632290) [2], and the last database has been recorded in the context of the “THINK2GO” clinical trial (clinicaltrials.gov, NCT06243952, no publication yet). All participants were implanted with either one or two WIMAGINE [16] implant(s) designed to record multi-channels ECoG signals with a sampling frequency of 586 Hz. Tab. 1 provides a description of each database.
In BCI–Exo5, the participant was trained to control a virtual exoskeleton. Neural features are classified among five states: right hand translation (AS_{RH}), left hand translation (AS_{LH}), right wrist rotation (AS_{RW}), left wrist rotation (AS_{LW}) and IS. In BCI–Exo8 the same participant was trained to control a virtual exoskeleton with eight states: the same five states plus left grasp (AS_{LG}), right grasp (AS_{RG}), and walk (AS_{RUN}). In BSI–1, the participant was trained to control an implanted electrical epidural stimulator to activate their own left leg and right leg to walk. Data were classified among three states: left hip (AS_{LHp}), right hip (AS_{RHp}) flexion and IS. In BSI–2, the participant had to control an implanted electrical epidural stimulator to use their own legs and walk similarly to the participant of the BSI–1 database.
During each recording session, an experimenter was next to the participant. The experimenter cued the participant to perform specific tasks (or to idle) through a software. The participant had to perform the mental task associated to the cued task, which is considered as the ground truth of motor intent for machine learning purposes. The number of trials per state was not fixed in advance, each state was repeated as many times as necessary until the experimenter judged that the model was effectively decoding the task in question.

Features extraction:
The Continuous Complex Wavelet Transform (CCWT) using the Morlet mother wavelet was used to compute neural features every 100ms. A decimation step was then applied to reduce the temporal dimensionality and set the sampling frequency to 10 Hz. The experimenter configured the frequency settings and the time length of the epoch (number of 100ms slices) thanks to the recording software.
The model's input neural features formed a tensor $X_t \in \mathbb{R}^{T \times F \times C}$ with T epoch duration (number of 100ms slices), F is the number of frequencies and C the number of channels.

Performance evaluation:
From the saved databases, we retrain models with different desired balance between the IS and AS. The desired balance ratio was encoded with the following vector $\tilde{r} = \left[\tilde{r}^{(IS)}\ 1\ \ldots\ 1\right]$, where $\tilde{r}^{(IS)}$ is the ratio of the IS among AS. Different values for $\tilde{r}^{(IS)}$ were tested: from 1 to 5 with a 0.5 step. The desired ratio vector of each state is then normalized to ensure the sum of ratio is equal to 1: $r = \tilde{r}/||\tilde{r}||$. The vector r is then used is equation (4-5).
Leave-one-out cross-validation was employed for each database, wherein all but one recording session were used for model training, and the held-out session was used for evaluation.
Models are trained in a pseudo-online configuration. The data are presented to the learning algorithm sequentially, as if the model was trained in real-time. The advantage of pseudo-online testing is that it enables the evaluation of a model without the necessity of conducting a real experiment with a user, thereby reducing both cost and time. The limitation of this approach is that it does not incorporate user feedback during the model's learning process as it does during online experiments. Four criteria are used to validate our approach, evaluating the decoder accuracy, false activations via the false positive rate, noise level in the prediction, and latency of the decoding.
The balanced accuracy ($BAcc$) metric is used to evaluate the model's ability to classify data correctly. Increasing $\tilde{r}^{(IS)}$ will favor the prediction of the IS in comparison with AS. It is necessary to ensure the algorithm does not degrade excessively the $BAcc$ while decreasing the number of false activations. This metric is insensitive to class imbalance in the testing set.
The false positive rate (FPR) measures how much the model predicts AS while the IS is the desired state. For this criterion, merge all the AS to form the positive class and the IS is considered as the negative class. By increasing $\tilde{r}^{(IS)}$, we expect the FPR to decrease and tend to 0. The FPR is given by: $\mathrm{FPR} = 1 - \mathrm{Rec}^{(IS)}$ where $\mathrm{Rec}^{(IS)}$ refers to the recall of the IS.
In addition, the noise level (NL) of the IS probability prediction is is evaluated. The NL is estimated by computing the Shannon entropy of the probability prediction. In the ideal case where the prediction probability is binary, the Shannon entropy of the predicted probability should

CC BY 4.0 - https://creativecommons.org/licenses/by/4.0/
This CC license does not apply to third-party material and content noted otherwise.

 10.3217/978-3-99161-093-9-005

Table 1: Description of databases.

Name of database	Number of implants	Implant(s) location	Motor impairment	Number of sessions	Mean duration of session	Recorded X months after implantation
BCI-Exo5	2	Sensorimotor cortex see [1], figure 2	Tetraplegia	8	46 min	16 to 17
BCI-Exo8				10	29 min	31 to 32
BSI-1	2	Sensorimotor cortex see [2], figure 1.b	Paraplegia	26	5 min	2 to 6
BSI-2	1	Sensorimotor cortex on the interhemispheric fissure	Paraplegia	24	14 min	1 to 4

be equal to zero. Noisy predictions lead to increase the Shannon entropy. We propose to modify the Shannon entropy function to estimate the NL of the IS:

$$NL = \frac{\log(2)}{e^{-1}} \times \frac{1}{T} \sum_{t=1}^{T} \tilde{Y}^{(IS)}_{t,y=\mathrm{IS}} \log_2 \left(\tilde{Y}^{(IS)}_{t,y=\mathrm{IS}} \right), \quad (6)$$

where $\tilde{Y}^{(IS)}_{t,y=\mathrm{IS}}$ is the probability prediction of the IS at the time iteration t when the current desired state is the IS. The modification of the Shannon entropy (mean instead of the sum) and adding the coefficient $\frac{\log(2)}{e^{-1}}$ bounds the criterion between 0 and 1, but keeps it equal to 0 when $\tilde{Y}^{(IS)}_{t,y=\mathrm{IS}}$ is binary. Note that the noise level is only computed for the samples corresponding to the desired IS, to evaluate the stability of the IS.

Additionally, the latency measures the delay between the cue given by the experimenter and the prediction of the cued state. The latency is defined as the time shift maximizing the cross correlation, given by:

$$Latency = \mathrm{argmax}_{\tau} \left(\sum_k E \left[Y^{(k)}_t \tilde{Y}^{(k)}_{t-\tau} \right] \right), \quad (7)$$

with $\tau > 0$.

The latency has an impact on the $BAcc$, the FPR and the NL. To compute these criteria, the prediction signal is shifted after computing the global latency to "align" the prediction signal with the target signal. The $BAcc$, the FPR and the NL are computed using the shifted prediction as in [17].

In addition to criteria evaluating the decoder performance, the class imbalance ratio (CIR) is provided to measure the class imbalance in the training dataset. The CIR is equal to the size of the majority class divided by the size of the minority class, taking into account the weights $w^{(k)}_u$ at each iteration [4]. A CIR equal to 1 means the training database has been perfectly balanced. It is expected that CIR tends toward $\tilde{r}^{(IS)}$ for all experiments.

CONFIGURATION

Tab. 2 gives the feature configuration for each database. The tensor size is given in the format $[T, F, C]$ with T epoch duration (number of 100ms slices), F is the number of frequencies and C the number of channels.

From the study on the HMM-RSWNPLS [4], we empirically chose a maximum weight equal to 10, to help the model balance the classes easily without having weights large enough to bias the model and degrade its performance.

Table 2: Feature settings.

Database	Tensor size	Frequencies (Hz)
BCI-Exo5	[10, 15, 64]	[10:10:150]
BCI-Exo8	[10, 15, 64]	[10:10:150]
BSI-1	[2, 24, 64]	[2 5:5:100 125 150 200]
BSI-2	[2, 29, 32]	[2 10:5:100 110:10:200]

A test on unused sessions from the BSI-1 database shows that the scaling parameter $\theta = 3$ decreases the latency by 33% and keeps the balanced accuracy stable (+0.08%). We fixed $\theta = 3$ for all the computations with all databases.

RESULTS

Fig. 1 displays the distribution of each metric. In each plot, the distributions are grouped by database and presented in ascending order of $\tilde{r}^{(IS)}$. The Mann-Whitney U-test is performed to evaluate the significance of between the first distribution ($\tilde{r}^{(IS)} = 1$) and the other distributions ($\tilde{r}^{(IS)} > 1$) except for the CIR.

The RSWNPLS succeeds to control the class imbalance for the BSI-1 and the BSI-2 for every ratio $\tilde{r}^{(IS)}$. For the BCI–Exo5, the RSWNPLS succeeds to control the class balance for $\tilde{r}^{(IS)} \leq 4$ (see Fig. 1C). However, for $\tilde{r}^{(IS)} > 4$, the RSWNPLS fails to make the CIR equal to $\tilde{r}^{(IS)}$. This observation is even more pronounced with the BCI–Exo8 where the RSWNPLS never reaches the expected CIR.

The mean $BAcc$ (Fig. 1A) tends to decrease as the ratio increases for all databases. The decrease is significant with regards to the Mann-Whitney U-test only after $\tilde{r}^{(IS)} = 3$. Medians and dispersion of the FPR (Fig. 1D) and the NL (Fig. 1B) appear to decrease exponentially as $\tilde{r}^{(IS)}$ increases. In three databases, the difference in these scores is already significant when $\tilde{r}^{(IS)}$ increases from 1 to 1.5.

The latency (Fig. 1E) decreases slightly when $\tilde{r}^{(IS)}$ increases. The median latency increases in the BCI-Exo5 database for $\tilde{r}^{(IS)} > 4$. Change in the latency distribution are significant with regards to the Mann-Whitney U-test only for $\tilde{r}^{(IS)} > 3$, except for the BCI-Exo8 databases.

The Fig. 2 illustrates two examples of prediction centered around one cue of the IS, for $\tilde{r}^{(IS)}$ equal to 1 and 3. These examples show the probability prediction of the IS is smoother when $\tilde{r}^{(IS)} = 3$. Also, there is less confusion between the IS and AS when the cue is IS. Note that some AS predictions are noisier since the model favors IS predictions.

CC BY 4.0 - https://creativecommons.org/licenses/by/4.0/
This CC license does not apply to third-party material and content noted otherwise.

 10.3217/978-3-99161-093-9-005

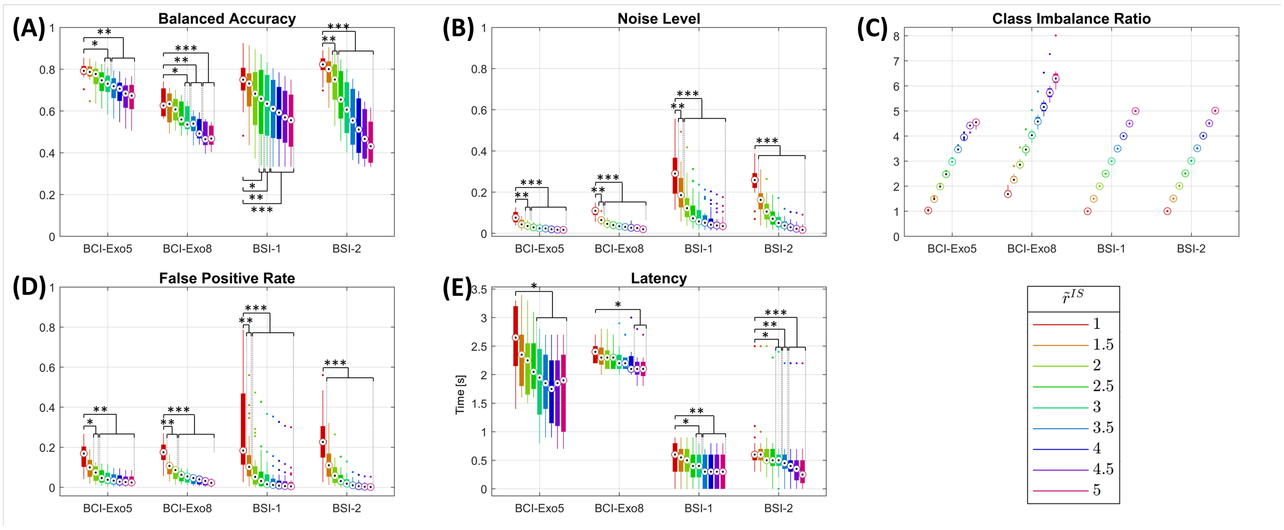

Figure 1: Performance criteria per database (**A**: *BAcc*, **B**: *NL*, **C**: *CIR*, **D**: *FPR* and **E**: *Latency*). In each plot, the four group correspond to the four databases. Boxplots are given for each $\tilde{r}^{(IS)}$ from 1 to 5. Note that for the *CIR* that the dispersion is very low, but multiple points are displayed. The Mann-Whitney U-test is performed between the first boxplot and each other to evaluate the hypothesis that the two distributions are different. Significative p-values are displayed (* when p<0.05, ** when p<0.01 and *** when p<0.001).

DISCUSSION

A primary goal of recent BCI development is to create home-use systems that can be used outside of a laboratory setting and without external assistance. Recent publications addressed the related issues. In particular, the HMM-RSWNPLS algorithm was proposed in [4], aimed to simplify real-time model training by automatically balancing the classes using sample weighting. Previously, experimenters had to manually ensure class balance during model training [1, 2]. The HMM-RSWNPLS relieves experimenters of this task representing a step toward unassisted BCI use. One limitation of the weighting strategy proposed in [4] is that the IS is treated equally with all other states, potentially increasing the *FPR* of the system. For home-use, unwanted activations of the BCI system can be particularly perturbing and even dangerous, depending on the effector activated without the user's intent. This study explores a new weighting strategy that favors the IS in model predictions and limits false system activations.

Test results show targeting class balance while training the decoder may provide better performance in asynchronous BCI compared to a perfect class balance ($\tilde{r}^{(IS)} = 1$). The algorithm successfully reaches the desired balance between classes in databases with low number of states. For a large number of states, the class balance tends to the desired balance, but sometime fails to reach it. Some states may be missing for multiple update iterations (because the experimenter was focusing on other states), which makes it impossible for the algorithm to balance missing states until they are trained again. The algorithm reduces the *FPR* and the *NL* of the IS. It also slightly reduces the latency. All of these improvements are made with a minimal loss in accuracy. A $\tilde{r}^{(IS)}$ between 1.5 to 2.5 seems to be a good trade-off to reduce the *FPR* with minor degradation of the *BAcc*. For example, $\tilde{r}^{(IS)} = 2$ degrades by 3.68% to 10.51% the *BAcc* and decreases the *FPR* by 55.36% to 70.08% in comparison with $\tilde{r}^{(IS)} = 1$. It also allows to have a model with a lower latency. Note that the optimal ratio IS/AS can vary from one user to another.

In this study, the proposed class balancing strategy was integrated into the HMM-RSWNPLS, but it may be adapted for any online training algorithm to train model in an incremental way. Additionally, the proposed algorithm could be tested on other signals recorded with other modality (*e.g.* EEG or intracortical EEG).

Model performance was evaluated using multiple criteria. To the best of our knowledge, there is no consensus in the literature about which criterion to use to evaluate the latency and the noise level of a prediction probability. Other criteria could be explored to evaluate the latency and the noise level. Multiple criteria analysis complicates decision-making. To address this issue, future studies will consider a multi-criteria decision analysis to facilitate the selection of the most appropriate hyperparameter value.

ACKNOWLEDGMENTS

We would like to thank all the members of the Brain Computer Interface project, composed of the multidisciplinary technical and clinical teams at Clinatec (CEA-LETI and University Grenoble Alpes), as well as the NeuroRestore team for conducting the experiments with the participants. This work has been supported by the European Commission (EIC 2021-TransitionChallenges-01-01 ReverseParalysis 101057450), Fonds Clinatec and its sponsors Malakoff Humanis, Covéa and Klésia (BCI4GRASP) and Institut Carnot Leti.

CC BY 4.0 - https://creativecommons.org/licenses/by/4.0/
This CC license does not apply to third-party material and content noted otherwise.

10.3217/978-3-99161-093-9-005

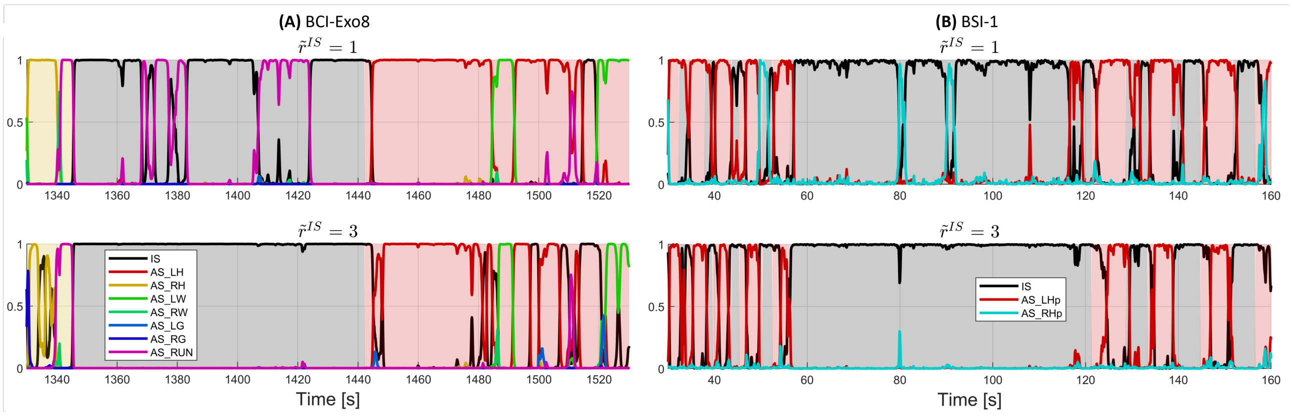

Figure 2: Example of predictions from BCI-Exo8 **(A)** and BSI-1 **(B)**. Background color show the desired states and the curves represent the predicted probability of each state. Top plots provide results for $\tilde{r}^{(IS)} = 1$ and bottom plots provide results for $\tilde{r}^{(IS)} = 3$.

REFERENCES

[1] Moly A et al. An adaptive closed-loop ECoG decoder for long-term and stable bimanual control of an exoskeleton by a tetraplegic. Journal of Neural Engineering. 2022;19(2):026021.
[2] Lorach H et al. Walking naturally after spinal cord injury using a brain–spine interface. Nature. 2023;618(7963):126–133.
[3] Patrick-Krueger KM, Burkhart I, Contreras-Vidal JL. The state of clinical trials of implantable brain–computer interfaces. Nature Reviews Bioengineering. 2024:1–18.
[4] Souriau R, Martel F, Aksenova T. Recursive sample weighted – N-way partial least squares for brain-computer interface decoder online learning with class imbalance. Biomedical Signal Processing and Control. 2026;113:109025.
[5] Zhang L, Li C, Zhang R, Sun Q. Online semi-supervised learning for motor imagery EEG classification. Computers in Biology and Medicine. 2023;165.
[6] Krell M, Wilshusen N, Seeland A, Kim S. Classifier transfer with data selection strategies for online support vector machine classification with class imbalance. Journal of Neural Engineering. 2017;14(2):025003.
[7] Tibrewal N, Leeuwis N, Alimardani M. Classification of motor imagery EEG using deep learning increases performance in inefficient BCI users. PLOS ONE. 2022;17(7):e0268880.
[8] Peksa J, Mamchur D. State-of-the-Art on Brain-Computer Interface Technology. Sensors. 2023;23(13):6001.
[9] Haotian X, Anmin G, Jiangong L, Fan W, Peng D, Yunfa F. Online adaptive classification system for brain–computer interface based on error-related potentials and neurofeedback. Biomedical Signal Processing and Control. 2023;82:104554.
[10] Ferdinand Q et al. ECoG-Based Movement Classification and Limbs 3D Translation Prediction : A Deep Learning Study. In: 2025 International Joint Conference on Neural Networks (IJCNN). Jun. 2025, 1–10.
[11] Souriau R et al. Hierarchical Hidden Markov Model for Online Decoding in Brain-Computer Interface. In: 2024 32nd European Signal Processing Conference (EUSIPCO). IEEE: Lyon, France, Aug. 2024, 1466–1470.
[12] Zhou Y et al. Shared Three-Dimensional Robotic Arm Control Based on Asynchronous BCI and Computer Vision. IEEE Transactions on Neural Systems and Rehabilitation Engineering. 2023;31:3163–3175.
[13] Yu Y, Liu Y, Jiang J, Yin E, Zhou Z, Hu D. An Asynchronous Control Paradigm Based on Sequential Motor Imagery and Its Application in Wheelchair Navigation. IEEE Transactions on Neural Systems and Rehabilitation Engineering. 2018;26(12):2367–2375.
[14] Zhang R et al. A BCI-Based Environmental Control System for Patients With Severe Spinal Cord Injuries. IEEE Transactions on Biomedical Engineering. 2017;64(8):1959–1971.
[15] Xuan H, Yang B, Li X. Exploring the impact of temperature scaling in softmax for classification and adversarial robustness. 2025. arXiv: `2502.20604 [cs.LG]`. [Online]. Available: `https://arxiv.org/abs/2502.20604`
[16] Mestais CS, Charvet G, Sauter-Starace F, Foerster M, Ratel D, Benabid AL. WIMAGINE: Wireless 64-Channel ECoG Recording Implant for Long Term Clinical Applications. IEEE Transactions on Neural Systems and Rehabilitation Engineering. 2015;23(1):10–21.
[17] Ferdinand Q et al. Ecog-based movement classification and limbs 3d translation prediction : A deep learning study. 2025 International Joint Conference on Neural Networks (IJCNN). 2025:1–10.

CC BY 4.0 - https://creativecommons.org/licenses/by/4.0/
This CC license does not apply to third-party material and content noted otherwise.