POMDP-BCI: A Benchmark of (re)active BCI using POMDP to Issue Commands

Juan J Torre Tresols, Caroline P C Chanel, Frédéric Dehais

POMDP-BCI: A benchmark of (re)active BCI using POMDP to issue commands

Juan J. Torre Tresols, Caroline P. C. Chanel, Frédéric Dehais

***Abstract*— Objective: Past research in Brain-Computer Interfaces (BCI) have presented different decoding algorithms for different modalities. Meanwhile, highly specific decision making processes have been developed for some of these modalities, while others lack such a component in their classic pipeline. The present work proposes a model based on Partially Observable Markov Decission Process (POMDP) that works as a high-level decision making framework for three different active/reactive BCI modalities. Methods: We tested our approach on three different BCI modalities using publicly available datasets. We compared the general POMDP model as a decision making process with state of the art methods for each BCI modality. Accuracy, false positive (FP) trials, no-action (NA) trials and average decision time are presented as metrics. Results: Our results show how the presented POMDP models achieve comparable or better performance to the presented baseline methods, while being usable for the three proposed experiments without significant changes. Crucially, it offers the possibility of taking no-action (NA) when the decoding does not perform well.Conclusion: The present work implements a flexible POMDP model that acts as a sequential decision framework for BCI systems that lack such a component, and perform comparably to those that include it. Significance: We believe the proposed POMDP framework provides several interesting properties for future BCI developments, mainly the generalizability to any BCI modality and the possible integration of other physiological or brain data pipelines under a unified decision-making framework.**

***Index Terms*— Brain-computer interface, EEG, POMDP, Active Perception**

I. Introduction

Brain-computer interfaces (BCI) are systems that enable the communication between the user and a machine without the direct intervention of motor pathways [1]. Active and reactive brain-computer interface (BCI) systems allow users to send commands to machines using different types of brain activity, such as imagined motor movements (MI) [2]), or exposition to certain types of stimuli (i.e., steady-state visual evoked potentials (SSVEP) [3] or other evoked potentials (e.g. P300) [4]) in response to specific stimuli. However, many BCI implementations suffer from limitations, such as the inability of the system to remain in an ”idle” or ”waiting” state and not take (a wrong) action. This is a crucial limitation that must be addressed in order to ensure that BCI systems are safe and effective for use in various applications [5]–[9]. In addition, these BCI implementations require the specification of several system parameters in advance, such as the length of the data window to be analyzed and any thresholding parameters required by the decoding algorithm, which must be set equally for all users. [10]–[13].

At a conceptual level, brain-computer interfaces (BCIs) enable user-machine communication by decoding the user's cerebral activity using brain imaging techniques such as electroencephalography (EEG) [14]. From this perspective, the BCI problem can be seen as an agent trying to obtain information about a hidden true state in the user's brain in order to determine which action or command needs to be performed by the machine.

Interestingly, the field of active perception presents a scheme that aims to reduce uncertainty about an unknown variable by collecting useful information from the environment [15]. According to [16], the goal of active perception is to collect observations to predict the values of unknown variables. In its simplest form, a brain-computer interface (BCI) can be viewed as an active perception scheme that collects observations from the user's brain and decides to wait if there is not enough information available to make a decision.

A popular approach for this paradigm is to formulate it as a sequential decision-making problem based on the Partially Observable Markov Decision Process (POMDP) [17]. POMDPs provide a versatile framework for decision-making in the face of uncertainties related to an agent's perception capabilities and the effects of its actions. From an active perception standpoint, POMDPs can enable an agent to decide how long to engage in perception (e.g., by taking information-acquisition actions) until sufficient information has been gathered to accurately identify a hidden state (e.g., by performing an identification action).

In this context, the objective of the current work is to evaluate an active perception POMDP-based paradigm on BCIs by implementing and testing a POMDP-BCI framework on three different active/re-active BCIs: SSVEP, Code-VEP (CVEP) and MI. Specifically, modeling aspects are presented and performance of the proposed framework is discussed and evaluated against classic decoding methods for each of these approaches. Then, current limitations and future research

Juan Jesús Torre Tresols is with the Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-SUPAERO), Université de Toulouse, Toulouse, France (e-mail: juan-jesus.torre-tresols@isae-supaero.fr).

Caroline P. C. Chanel and Frédéric Dehais are with ANITI chair ”neuroadaptive technology”, and with the Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-SUPAERO), Université de Toulouse, Toulouse, France (e-mail: name.surmane@isae-supaero.fr).

This work was supported in part by the Artificial and Natural Intelligence Toulouse Institute (ANITI) - ANR-19-P3IA-0004.

directions of such a POMDP-BCI approach are addressed.

The rest of the paper is structured as follows: First, section II will discuss previous research on POMDP applied to active perception, POMDP applied to human-machine interaction, and finally POMDP applied to BCIs. Then, Section III contains detailed explanations of the datasets employed, followed by the POMDP model definition and explanation. The results section reports the accuracy, number of false positives, the decision time and the number of trials where the POMDP decides not to act (called "no-action" trials or NA) for all the different BCI approaches. Finally, the discussion section will comment on the results and discuss limitations and future directions of the proposed framework.

II. Related Work

A. POMDP and Active Perception

Although active perception problems have been addressed by numerous probabilistic planning frameworks, a number of studies have formalized such a problem using the POMDP framework [18]–[22], which allows to plan actions (e.g. controls) that maximize a long-term objective (i.e., a non-myopic course of actions) in environments where selecting the most relevant sources of information is crucial. In particular, Guo et al. (2003) [22] proposed a method to represent an active sensing problem as a POMDP, where each state of the model corresponds to a different class of objects to be classified. This approach is similar to the one used in the BCI problem, which is described in more detail in section III-A.

Another relevant application of POMDP for active sensing is when there are numerous information sources available to the agent, and it must decide which ones to query in order to obtain the maximum relevant information given its long-term objectives. In [18], a robot had to navigate a maze in order to arrive at a goal and avoid obstacles. Meanwhile, several camera-equipped drones patrolled the area in fixed concentric routes. The agent then had to choose which drones to query in order to obtain the most relevant information from the environment to navigate efficiently.

Other works proposed POMDP model extensions to active perception applications like ρPOMDP [23], a model that evaluates courses of actions by the expectation of belief-dependent rewards[1], prioritizing information acquisition, and POMDP with informed rewards (POMDP-IR) [24], which allows the user to reward the agent for reaching a desired level of certainty about a particular state variable.

Although these more refined POMDP approaches could be potentially useful for BCI problems in the future, the present work focuses on a POMDP implementation using the classical POMDP model where the information-collecting actions consist on *waiting on* more data (i.e. for new observations). In the problem that we describe in Sec. III-A, the actions of the agent do not influence the world, and thus the problem can be resumed in how much information is needed before confidently identifying a hidden state (i.e. a BCI target state), which can be modeled as a classical POMDP. Note, it conceptualy differs from POMDP-IR case as no preference between states exists and no desire level (or threshold) of certainty is a priori defined. The goal is to translate any recognized brain activity to actions correctly (whatever they are).

[1]More specifically, defining a reward function that is belief-dependent instead of state-dependent as in the classical POMDP model.

B. POMDP and Human-Machine Interaction

Although not using brain data, there is a body of literature on the use of POMDP models for human-machine interaction, using other physiological signals. Research has been conducted modeling collaboration or adaptive interaction frameworks between user and machine using Mixed-Observability Markov Decision Processes (MOMDP) or POMDPs in search and rescue [25], target search [26] scenarios and pilote assistive systems [27]. These studies incorporated uncertainties associated with the actions of human operators on the interface, as well as uncertainties about their cognitive state (such as cognitive availability and mental workload) into the POMDP model that governs the interaction. This enables the belief state of the POMDP to continuously monitor both the human state and other mission-related partially observable state variables.

C. POMDP and BCI

To the knowledge of the authors, only two previous research works have used POMDPs to model the BCI-related decision problem. In [28], the authors propose a POMDP to classify SSVEP stimuli. In this study, the SSVEP spectral features are extracted using the Fast Fourier Transform (FFT) algorithm, and used directly as observations of the POMDP model. Given the continuous nature of those POMDP observations (and the resulting difficulty of solving a such POMDP problem), they proposed an approach to discretize the feature space (observation space) in order to obtain a discrete set of observations. The discretization procedure was necessary due to the direct use of spectral features. This study was conducted prior to the introduction of state-of-the-art SSVEP classification methods such as Canonical Correlation Analysis (CCA) [12] and Task-Related Component Analysis (TRCA) [11], which enable working directly with classification labels instead of continuous spectral scores. These advanced methods eliminate the need for such a complex discretization procedure, however they are specific to SSVEP, making them not generalizable to other BCI modalities.

In [29], the POMDP model controls both the selection of the target as well as the order of the stimulus presentation in a P300 speller paradigm. The authors shown how POMDP can manage BCI where the optimal time of stimulation (or number, in their case) is not a priori known for different users. Their results suggested that, in the case of P300 speller, controlling the order of stimulation provided an improvement in time with respect to the classic random stimulation pattern. Other reactive BCI modalities (e.g. SSVEP), present all possible targets simultaneously, so there is no need for controlling the stimulus presentation using the POMDP model, and then, the decision-making problem climbs into an active perception problem.

With these previous work in mind, we believe a revision of the work on [28] is required since SSVEP literature has seen notable contributions in both decoding algorithms [11], [13]

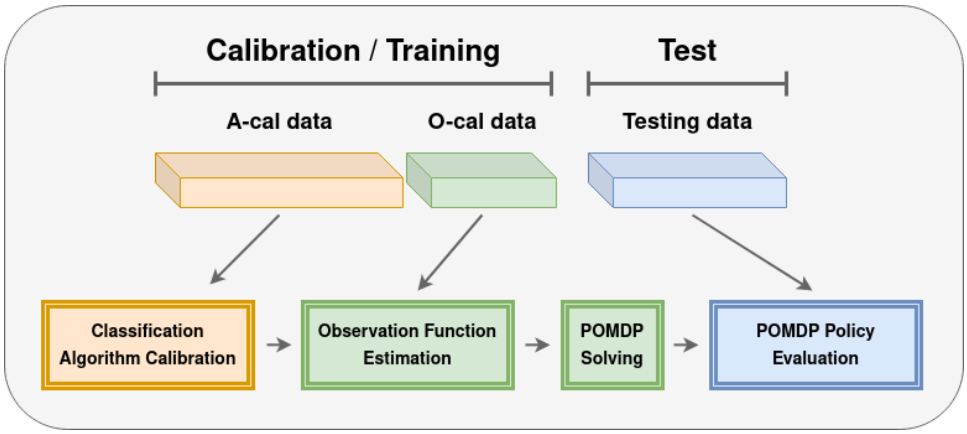

Fig. 1. Division applied to EEG data (top) and pipeline steps (bottom) for each experiment described. Data is divided into that used to train the classification algorithm ('A-cal', orange), that used to estimate the POMDP observation function ('O-cal', green), and that used to test the POMDP ('Testing', blue)

and stimulus presentation [30], [31]. In detail, we choose to exploit these existing approaches to obtain the POMDP-BCI model. Specifically, our approach allows for the construction of a confusion matrix (e.g. using a calibration step), representing the performance of these decoding algorithms, to be used as the observation function of the POMDP. The principal advantage is that one can define a personalized POMDP-BCI model (i.e. using its specific observation function) *on the fly* without the need for any other threshold or parameter tuning. Moreover, we employ our POMDP-BCI framework to other (re)active BCI modalities that we believe were not yet explored in literature, such as CVEP and MI.

III. Methods

In this section, we will describe the three experiments we have conducted with three different active/reactive BCI modalities: SSVEP (Experiment 1), CVEP (Experiment 2) and MI (Experiment 3). Before describing each dataset in detail, a general introduction for POMDP-BCI approach is presented, followed by the specificities of each experiment.

A. POMDP-BCI model

1) Recall on POMDPs: A POMDP is formally defined as a tuple $\langle S, A, \Omega, T, O, R, \gamma, b_0 \rangle$, where: S is the set of states; A is the set of actions; Ω is the set of observations; $T : S \times A \times S \rightarrow [0; 1]$ is the transition function denoting the probability $T(s', a, s) = p(s'|a, s)$ of reaching state $s' \in S$ given the action $a \in A$ is performed in state $s \in S$; $O : \Omega \times S \rightarrow [0; 1]$ is an observation function such that $O(o, s) = p(o|s)$, denoting the probability of observing $o \in \Omega$ given state $s \in S$; $R : S \times A \rightarrow \mathbb{R}$ is the reward function associating a real value for each state-action pair; γ is the discount factor; and b_0 is the initial probability distribution over states such that $b_0(s) = p(s_0 = s)$, with $b_0(s) \in \Delta$, the belief state space.

At each decision time step t, the agent takes an action $a \in A$, receives an observation $o \in \Omega$ and updates its *belief state* $b \in \Delta$ using the Bayes' rule:

$$b_a^o(s') = \eta^{-1} p(o|a, b) = \eta^{-1} p(o|s') \sum_{s \in S} p(s'|s, a) b(s) \quad (1)$$

where $\eta = \sum_{s'} p(o|s') \sum_s p(s'|s, a) b(s)$ is a normalization factor. The belief state can be seen as a complete information state (i.e. history), as it concatenates all the action-observation sequences such as $b_t(s) = p(s_t = s|o_t, a_{t-1}, o_{t-1}, ..., a_0)$.

The goal of solving a POMDP is to find a policy $\pi : \Delta \rightarrow A$, which maximizes a performance criterion, the so-called Value Function, which is generally defined as the expected discounted sum of rewards:

$$V^\pi(b) = \mathbb{E}^\pi \left[\sum_{t=0}^{\infty} \gamma^t r(b_t, \pi(b_t)) \middle| b_0 = b \right]. \quad (2)$$

Thus, the optimal policy π^* is the policy that maximizes the value function (Eq. 2). It can be shown that V^{π^*} satisfies the Bellman equation ($V^{\pi^*} = V^*$):

$$V^*(b) = \max_{a \in A} \left[r(b, a) + \gamma \sum_{o \in \Omega} p(o|a, b) V^*(b_a^o) \right], \quad (3)$$

where $r(b, a) = \sum_{s \in S} R(s, a) b(s)$. As $r(b, a)$ can be see as the average immediate gain. The optimal policy π^* can be extracted from Eq. 3 when it converges for all belief states.

In practice, the exact optimal solution of an infinite horizon POMDP is hard to be achieved. However, several algorithms allow to approximate a near-optimal value function and a near-optimal policy using iterative methods. In our POMDP-BCI methodology, we used one of such algorithms, called SARSOP [32]. SARSOP is a well known POMDP solving algorithm which achieves a near-optimal policy solution in reasonable time, enabling us to exploit it *on the fly*.

2) POMDP modeling proposal: The POMDP model we used is the same for all three experiments. In each case, the state and observation spaces $S : \{s_1, \ldots, s_N\}$ and $\Omega : \{o_1, \ldots, o_N\}$ are made of N states and observations, respectively, with N being equal to the number of available BCI commands in the system, as well as the number of possible classification outputs. For example, an SSVEP BCI with 12 possible commands (i.e. $N = 12$), such as that used in experiment 1 (see section III-B.1), would be mapped to a POMDP model with 12 possible states and observations. For experiments 2 and 3, the same applies with $N = 11$ and $N = 2$, respectively. Similarly,

the POMDP action space $A : \{a_1, \ldots, a_N, a_w\}$ contains one action per state, denoted $a_1, \ldots, a_N$, to decide that the user is issuing each of the possible N commands, plus a *wait* action denoted by a_w, for when the model needs to collect more observations before taking a decision.

The observations for the model are the possible outputs from a regular classification algorithm, that varies depending on the experiment. The observation function O denoting the probability $p(o|s')$ of obtaining a given observation in a given next state is obtained by calculating a confusion matrix $F \in \mathbb{R}^{N \times N}$ of the trained classifier using empirical data. Assuming the rows of the confusion matrix represent the true N classes and its columns represent the N predicted classes, we can express the probability function as $p(o|s') = F_{s',o}$. Thus, for each experiment, data was divided into three parts (see Fig. 1): A-cal (orange), O-cal (green), testing (blue). A-cal data was used to train the corresponding algorithm, O-cal data was used to empirically estimate the POMDP observation function and to instantiate the POMDP model, and testing data was used to simulate the POMDP policy.

Due to the empirical approximation using limited data, we know that the approximated observation function $\hat{O}$ may not perfectly reflect the true probability of $p(o|s')\ \forall s' \in S, o \in \Omega$. Thus, we adopted a similar approach to the one applied by [29] and [27]. In order to account with this empirical error, we merged the obtained observation function $\hat{O}$ with a uniform distribution over states using a weighted average, as follows:

$$\bar{O}(s,a) = (1 - \omega_O)\hat{O}(s,a) + \omega_O \frac{1}{N} \tag{4}$$

with $\hat{O}(s,a)$ being the observation function obtained from data and ω_O being the weighting factor.

For the transition function T, the transition probabilities for the action *wait* a_w, (i.e. $T(s, a_w, s')$) was set as an identity matrix, defined as 1 when $s = s'$, and 0 otherwise. Note that we assume the BCI user will not change targets within a trial. However, for any other action $a_1, \cdots, a_N$ (i.e. identification actions), the transition probabilities were set as the uniform distribution over states. We assume it since we have no *a priori* information of what the next target state the BCI user will select after a command.

For the reward function R, we defined the reward of a_w to be equal to -1 for all states, incurring a small cost for the information gain. For all control actions (state identification action), the reward $R(s_n, a_n),\ \text{with}\ s_n = a_n$, was set to 10, rewarding the model for selecting the control action that matched the true state, whereas the reward for $R(s_m, a_n), \forall m \neq n$ incurs a penalty, i.e. a high cost for incorrect control actions. In experiments, different costs (i.e, negative rewards) were tested in order to assess the effect of different costs on false positives. The details about such costs are explained in the section corresponding to each experiment.

In order to acquire observations, a slice of EEG data is fed into the corresponding classifier, and the resulting output is given to the POMDP as an observation. For every experiment, we defined a data window length for all the observations, that is shifted in time by the amount of the time step.

The initial belief b_0 was set to the uniform distribution. In each trial, the model would perform an action, receive an observation and update the belief state as described above. Similarly to other approaches that implemented variable trial length in BCIs [33], we defined a maximum time for each trial (which varied depending on the experiments). If the POMDP-BCI model does not reach a state identification decision by the last time step, no other decision is taken and the trial was ended, which set the belief again to the uniform distribution.

Since both the maximum trial time and the time step are known, the maximum number of time steps per trial d_{max} is also pre-defined for each experiment. This allowed us to set the discount factor γ dynamically as: $\gamma = g^{1/d_{max}}$, with g being the desired value for the discount factor at the last time step (i.e. $\gamma^{d_{max}}$). This allowed for faster POMDP solving by relaxing the effect of how far in the future expected rewards weight on the initial belief state.

B. Experiments, Datasets and Evaluation Metrics

All three experiments were conducted using public available EEG datasets. Datasets for experiments 1 and 3 were downloaded directly using the 'MOABB' [34] Python package[2]. The POMDP implementation for the three problems is identical and was developed using the 'pomdp-py' [35] Python package[3]. All the code developed to carry on the analyses described in this paper, as well as the code used to generate all figures and tables (except Fig. 1) are available on Github[4].

As explained in section III, data was divided into three (see Fig. 1) for all experiments: A-cal data (Fig. 1, orange), which is the data used to train the underlying classification algorithm, O-cal data (Fig. 1, green), used to compute the confusion matrix and instantiate the POMDP model, and testing data (Fig. 1, blue). To avoid confusion with other terms (e.g., observation, calibration), we will refer to these divisions of the data with the aforementioned names, leaving the use of the word 'calibration' to denote the complete process of preparing for the testing phase (i.e. classification algorithm training plus POMDP instantiation and solving).

For all experiments, the POMDP decision step was 0.1s, meaning that the model takes an action (which includes not acting) every 0.1 seconds. This means that the EEG data fed to the classifier at each time step uses a sliding window whose step is equal to the POMDP's decision step. The length of the sliding window varies depending on the experiment. There is as well a maximum trial time. Consequently, the maximum number of time steps per trial is known for each experiment.

In all experiments, our POMDP-BCI approach was compared against the baseline, i.e. using the classification algorithm alone.

For all POMDP models (across experiments and participants), the parameter ω_O (see Eq. 4) was set to 0.3, as in [29]. Similarly, the parameter g that defines the desired value of the discout factor γ at the last time step of each trial was set to 0.25.

[2]Available on `https://github.com/NeuroTechX/moabb`
[3]Available on `https://github.com/h2r/pomdp-py`
[4]`https://github.com/neuroergoISAE/POMDP-BCI`

The performance evaluation in all experiments and methods was based on several metrics, including accuracy, false positives (FP), average decoding time, and, when applicable, the number of no-action (NA) trials. It is worth noting that all classifications, both in the POMDP approach and the baseline methods, were conducted using the inter-trial data from each dataset.

In the case of methods that employ single-time classification, such as the baseline method in experiment 1 and experiment 3, false positive (FP) trials are defined as trials where the predicted label differs from the true label. The average decoding time for these methods is equal to the length of the data window used for classification.

On the other hand, methods that perform multi-time prediction using multiple data windows, such as the POMDP approach and the baseline method in experiment 2, may encounter situations where a decision is not reached even after processing all available data within a trial. In such cases, these trials are classified as no-action (NA). False positive (FP) trials, for these methods, refer to trials where a decision is made, but the predicted label does not match the true label.

To calculate the average decoding time for participants using these latter methods, we compute the arithmetic mean of the decoding time across all test trials.

Additionally, and in order to compare the number of FP and average decoding time between POMDP and their respective baselines, statistical tests were performed. Firstly, normality and homoscedasticity assumptions were verified on all FP and avg. decoding time results. Since these assumptions were only respected for the FP scores of experiment 3, non-parametric Friedman and Nenmeyi post-hoc tests were used in all statistical analyses for comparison of effect sizes between experiments.

1) Experiment 1: The first experiment was conducted using the dataset described in [10]. The dataset consists in 12 SSVEP flickers, at frequencies between 9.25 and 14.75Hz acquired from 10 subjects. The data amounts to 15 blocks with one trial per class, for a total of 180 SSVEP trials. The algorithm used for this experiment was ensembled Task-Related Component Analysis (TRCA) [11]. TRCA uses a body of calibration data to compute a series of spatial filters that maximize the correlation between examples of the same class while minimizing the correlation of examples of different classes. It also averages all the calibration trials of a given class to create a 'template' for that class. When new data is obtained, a correlation score is calculated between the new example and the templates (using the pre-computed spatial filters). The class that obtains the highest correlation with new example is selected.

The EEG data window for this experiment was 0.5s (i.e., all POMDP observations are obtained using 0.5s data windows), and the maximum trial time was set to 1s (i.e. there is a maximum of 6 decision steps between 0.5s and 1s, each with 0.1s of interval). For the whole calibration process, which is composed by the TRCA calibration and observation function computation, a cross-validation was used to find the best combination of data for TRCA calibration and observation function estimation for each participant. Specifically, the TRCA models were trained using data from 8 experimental blocks, and the observation matrix was approximated using data from 5 experimental blocks. The POMDP testing was then performed on the remaining 2 blocks of data (i.e., 24 trials). Due to the template-matching nature of the TRCA algorithm, a different TRCA model was trained for each time step (i.e. from $t = 0$ to $t = 0.5$, from $t = 0.1$ to $t = 0.6$, etc.), and the final confusion matrix was obtained averaging the predictions for all time steps. For this experiment, different values of penalty (i.e. a negative reward when $R(s_m, a_n), \forall s_m \neq a_n$) were tested in the POMDP model. In detail, each participant had two different POMDP models with costs 100 and 1000, respectively. The reward for a correct identification (i.e. $R(s_n, a_n),$ with $s_n = a_n$) was not changed.

For baseline comparison, all data corresponding to calibration and observation function computation (i.e., 13 blocks of data) were used to train two TRCA models with different epoch lengths: 0.5s and 1s (resp. TRCA-0.5s and TRCA-1s). The accuracy on flickers identification of the POMDP-based and both TRCA baseline models was compared.

2) Experiment 2: The second experiment was conducted on C-VEP data using the dataset from [36][5]. The CVEP response is evoked by regulating visual stimuli using pseudo-random and aperiodic sequences, instead of the frequency-modulated pattern used in SSVEP [36], [37].

The dataset consists on 11 flickers acquired from 11 subjects. The dataset contains 15 blocks with one trial per class, for a total of 165 CVEP trials. The underlying algorithm for classification was the Convolutional Neural Network (CNN) used in [36], based on the architecture presented by EEG2Code [37]. The EEG2Code model proposes a neural network that takes EEG activity as input and outputs the predicted stimulation code that corresponds to that EEG data. Once the code is obtained, different methods are used to compare the predicted code with the different stimuli in order to select that which is being attended by the user.

Similarly to Experiment 1, the window length (i.e., length of the data used to obtain all observation) for our POMDP approach was set to a minimum of 0.5s and a maximum of 1s, resulting in a maximum of 6 time steps. The distribution of the data between calibration, observation and testing data was also identical to that of Experiment 1 (Note: Since this experiment has 11 classes, the total number of test trials is 22). Since the CNN is trained to regress the stimulation code from EEG activity, only one network is required, no matter the specific time window used. Similar to the Experiment 1, each participant had two different POMDP models with costs 100 and 1000, respectively. The reward for a correct identification did not change.

For the comparison against the baseline, each participant had the testing data classified using the same CNN trained for the POMDP using two different methods used in previous CVEP publications [36], [37]. These methods involve starting with a small window of data, which was set to 0.5s (30 frames at 60 frames per second), and performing multiple classifications on a data window of increasing length up to

[5]available on `https://doi.org/10.5281/zenodo.7277151`

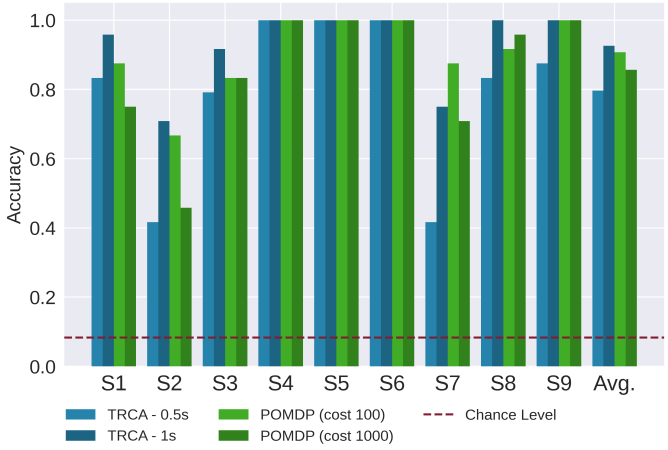

Fig. 2. Average accuracy per subject for Experiment 1 (TRCA) showing, from left to right: baseline (TRCA) at 0.5s of data, baseline (TRCA) at 1s of data, POMDP with cost 100 and POMDP with cost 1000.

1s (contrary to POMDP, which uses a fixed window length). The methods are:

- p-value based method: at each time step, the code obtained from the CNN is compared with the corresponding code of each target stimulus at the same time window using Pearson correlation. If the value of the most correlated code is statistically significant (with a p-value pre-defined by the experimenter), the target is selected, otherwise, the operation is repeated at the next time step. Each decision increased the data by 0.05s (corresponding to 3 frames at 60 frames per second), for a total of 10 decision steps.The p-value for this experiment was set to 0.001
- accumulation method: at each time step, the code obtained is compared with all the stimuli as in the p-value method. The one with the highest correlation is stored and the operation is repeated at each time step. Each decision increased the data by 0.016s (corresponding to 1 frame at 60 frames per second), for a total of 30 decision steps. If a certain number of predictions corresponds to the same stimuli, it is selected. The number of necessary classifications is decided beforehand by the experimenter. For this experiment, the number of necessary predictions of the same class was set to 15.

3) Experiment 3: The third experiment was conducted using the MI dataset used in the fourth BCI competition [38], comprising data from 9 subjects performing four different MI tasks: right hand, left hand, both feet and tongue. The data was acquired over two sessions with six runs each, for a total of 12 runs. Each run comprised 12 trials per class. Only the right hand and left hand trials were used for this experiment (24 trials per runs). The algorithm used as underlying classifier was a logistic regression using tangent space features extracted based on Riemannian geometry (RG+LR), see [39], [40] for more details on the classification process.

The data window length used for the POMDP was 3s, with a maximum of 4s. Regarding the division of the data, the whole first session was used for algorithm calibration, while the first three runs of the second session were used for the observation function estimation (i.e. estimating an inter-session classification performance) and the remaining three runs were used as testing data (i.e., 72 test trials). As with Experiments 1 and 2, each participant had two different POMDP models with costs 100 and 1000, respectively, and the reward for a correct identification did not change.

For comparison purposes, the baseline used the classification algorithm alone using static epochs of both 3s and 4s of length, and evaluated on the same testing data used for the evaluation of POMDP-based approach. Due to the inter-session nature of the classification process addressed in this experiment, adding the observation data (second session) to the training data (first session) for the baseline algorithm (similarly to Experiment 1) would introduce information about the second session, which invalidates the purpose of analysis.

IV. Results

For each experiment, we present the accuracy score for each participant for their POMDP model, as well as the two corresponding baselines. Additionally, the same information is shown in a separate table, which also includes information about the number of false positives (FP), average decision time and the number of times the system decided not to act (NA) when applicable. The tables also include the general performance in each of the aforementioned metrics (i.e. averaged across all participants).

TABLE I
EXPERIMENT 1 (SSVEP BCI) RESULTS (TEST SIZE: 24 TRIALS)

	TRCA (0.5s)			TRCA (1s)			POMDP (cost 100)				POMDP (cost 1000)			
Sub.	Acc	FP	Avg. Time	Acc	FP	Avg. Time	Acc	FP	Avg. Time	NA	Acc	FP	Avg. Time	NA
1	0.83	4	0.5	0.96	1	1.0	0.88	2	0.74	1	0.75	2	0.87	4
2	0.42	14	0.5	0.71	7	1.0	0.67	4	0.82	4	0.46	0	0.96	13
3	0.79	5	0.5	0.92	2	1.0	0.83	4	0.63	0	0.83	1	0.75	3
4	1.0	0	0.5	1.0	0	1.0	1.0	0	0.6	0	1.0	0	0.71	0
5	1.0	0	0.5	1.0	0	1.0	1.0	0	0.6	0	1.0	0	0.7	0
6	1.0	0	0.5	1.0	0	1.0	1.0	0	0.5	0	1.0	0	0.5	0
7	0.42	14	0.5	0.75	6	1.0	0.88	3	0.75	0	0.71	3	0.87	4
8	0.83	4	0.5	1.0	0	1.0	0.92	2	0.62	0	0.96	0	0.75	1
9	0.88	3	0.5	1.0	0	1.0	1.0	0	0.63	0	1.0	0	0.72	0
Avg.	**0.8**	**4.9**	**0.5**	**0.93**	**1.8**	**1.0**	**0.91**	**1.7**	**0.65**	**0.6**	**0.86**	**0.7**	**0.76**	**2.8**
(%)	–	**20**	–	–	**8**	–	–	**7**	–	**3**	–	**3**	–	**12**

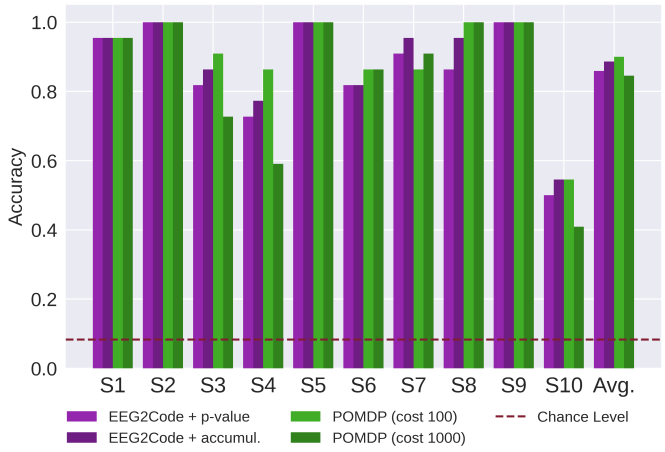

Fig. 3. Average accuracy per subject for Experiment 2 (C-VEP) showing, from left to right: EEG2Code + p-value, EEG2Code + accumulation, POMDP with cost 100 and POMDP with cost 1000.

A. Experiment 1 (SSVEP)

The average accuracy for each subject can be seen in Fig. 2, where the POMDP-100 scores in between TRCA at 0.5 and 1s. Notably, we can see than subject 2 (S2) and 7 (S7) performed under 0.5 accuracy with 0.5s TRCA, and the POMDP got comparable accuracy (S2) and even better (S7) than 1s TRCA. On the other side, most subjects perform equally or worse in POMDP-1000.

Table I shows the general (i.e. across subjects) accuracy of each approach (TRCA-0.5 = 0.8, TRCA-1 = 0.93, POMDP-100 = 0.91, POMDP-1000 = 0.86). It also shows that both POMDP reduces the number of FP when compared with TRCA-0.5 from 4.9 to 1.7 (POMDP-100) and 0.7 (POMDP-1000), and does the same for TRCA-1 (avg. FP = 1.8) with the exception of two subjects (sub. 3 and sub. 8) for POMDP-100, resulting in a comparable overall performance. Regarding statistical differences, Friedman test disclosed a main effect of the type of method on the FP ($Q = 13.41, p < 0.05, W = 0.49$). Nenmeyi post-hoc analyses only revealted a significant difference between POMDP-1000 and TRCA-0.5 ($p < 0.05$).

The POMDP approach offers the possibility of not acting (i.e. NA or 'No-action'), where TRCA always outputs an action. Two number of subjects who present NA trials is two and five, for POMDP-100 and POMDP-1000, respectively.

Regarding the average response time for each approach, both POMDP varies widely across participants. Those with perfect accuracy in TRCA are between 0.5 and 0.6 seconds, while the rest vary depending on how well TRCA performed. Crucially, the POMDP model adapted the length of the trial to each participant depending on the quality of the TRCA detection, resulting in performance comparable to 1s TRCA in an average of 0.65s for POMDP-100 and 0.76 for POMDP-1000. Statistical tests disclose a main effect of the type of method on the avg. decoding time using Friedman($Q = 25.884, p < 0.001, W = 0.95$), and Nenmeyi post-hoc tests reveal a significant difference between POMDP-100 and TRCA-1 ($p < 0.05$), as well as between TRCA-0.5 and POMDP-1000 ($p < 0.05$).

B. Experiment 2 (C-VEP)

Fig. 3 shows the mean accuracy for each subject for POMDP with costs 100 (POMDP-100) and 1000 (POMDP-1000), as well as the chosen baselines. POMDP-100 performs comparably or better than both baselines for all subjects except one (S7), while in POMDP-1000 most subjects keep the same performance, and others perform lower than baseline.

The average across subjects shown in Table II reveals that both POMDPs perform comparably to both CVEP baselines. The best accuracy is obtained by POMDP-100 (0.9), followed by the accumulation baseline (0.89), p-value baseline (0.86), and POMDP-1000 (0.85). For FP, all approaches perform similarly, with POMDP-1000 having the lowest average FP (0.4) across subjects, followed by p-value baseline (0.5), accumulation baseline (1.0) and finally POMDP-100 (1.3). The Friedman test did not disclose any effect of type of method on the FP ($Q = 4.26, p = 0.235, W = 0.142$). This is similar in the case of NA trials, with scores of 3 (POMDP-1000), 2.6 (p-value baseline), 1.2 (accumulation baseline) and 0.9 (POMDP-100).

Finally, the fastest method of the four is the p-value baseline with an avg. time across subjects of 0.56s, followed by POMDP-100 at 0.68s. The accumulation baseline and POMDP-1000 perform similarly in that regard with 0.76s and 0.79s, respectively. The Friedman test disclosed a main effect

TABLE II
EXPERIMENT 2 (C-VEP BCI) RESULTS (TEST SIZE: 22 TRIALS)

	CVEP (p-value)				CVEP (accumulation)				POMDP (cost 100)				POMDP (cost 1000)			
Sub.	Acc	FP	Avg. Time	NA	Acc	FP	Avg. Time	NA	Acc	FP	Avg. Time	NA	Acc	FP	Avg. Time	NA
1	0.96	0	0.52	1	0.96	0	0.75	1	0.96	1	0.63	0	0.96	1	0.73	0
2	1.0	0	0.5	0	1.0	0	0.75	0	1.0	0	0.6	0	1.0	0	0.71	0
3	0.82	0	0.62	4	0.86	0	0.76	2	0.91	2	0.75	0	0.73	0	0.86	6
4	0.73	3	0.55	3	0.77	3	0.77	2	0.86	1	0.78	2	0.59	0	0.9	9
5	1.0	0	0.5	0	1.0	0	0.75	0	1.0	0	0.61	0	1.0	0	0.71	0
6	0.82	0	0.63	4	0.82	1	0.78	2	0.86	2	0.73	1	0.86	2	0.84	1
7	0.91	1	0.57	1	0.96	0	0.76	0	0.86	3	0.66	0	0.91	0	0.80	2
8	0.86	0	0.53	3	0.96	0	0.75	1	1.0	0	0.61	0	1.0	0	0.74	0
9	1.0	0	0.51	0	1.0	0	0.75	0	1.0	0	0.6	0	1.0	0	0.7	0
10	0.5	1	0.64	10	0.55	6	0.78	4	0.55	4	0.85	6	0.41	1	0.95	12
Avg.	**0.86**	**0.5**	**0.56**	**2.6**	**0.89**	**1.0**	**0.76**	**1.2**	**0.9**	**1.3**	**0.68**	**0.9**	**0.85**	**0.4**	**0.79**	**3.0**
(%)	–	**2**	–	**12**	–	**5**	–	**5**	–	**6**	–	**4**	–	**2**	–	**14**

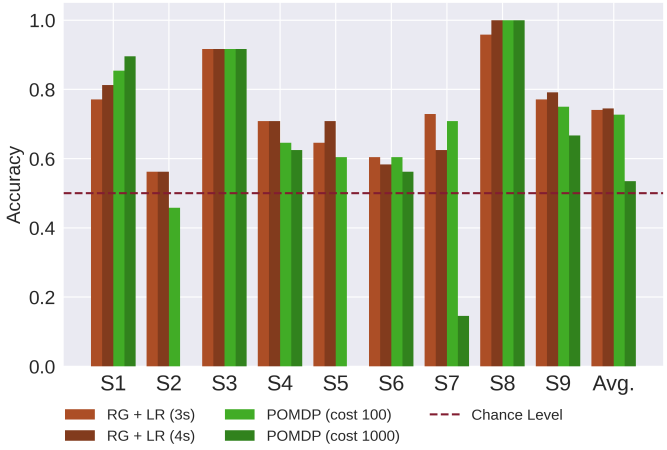

Fig. 4. Average accuracy per subject for Experiment 3 (MI) showing, from left to right: baseline (RG + LR) using 3s of data, baseline using 4s of data, POMDP with cost 100 and POMDP with cost 1000.

of the method on the avg. decoding time ($Q = 23.88, p < 0.001, W = 0.79$), and Nenmeyi post-hoc tests revealed a significant difference between the p-val method and POMDP-1000 ($p < 0.05$).

C. Experiment 3 (MI)

The average accuracy for each subject is presented in Fig. 4 for both POMDP-100 and POMDP-1000, together with the two baselines. POMDP-100 performs comparatibly to both baselines, which perform almost equally to each other. For POMDP-1000, most subjects show comparative performance to POMDP-100, however, three subjects (S2, S5 and S7) present a sharp drop in accuracy. It is shown in Table III that these subjects show close to none FP, and all (S2, S5) or almost all (S7) trials are NA trials. Contrary to Experiment 2, where the increased POMDP cost obtains comparable accuracy with less FP and more NA trials, here the increase in cost makes the POMDP model to decide to not act in almost every trial.

Because of this, we show the confusion matrix of S2 in Fig. 5, where we can see how poor classification score is achieved in one of the two classes with the underlying algorithm. In this case, the POMDP model chooses to not act due to high uncertainty avoiding the high cost of 1000, and risks more false positives at the low cost of 100. Considering this together with the fact that this is a 2-class problem (i.e. the chance level is 0.5), it is likely that at least part of the correctly classified trials for both POMDP-100 and the two baselines can be attributed to chance. Even considering that, POMDP-100 achieves lower FP across subjects than both baselines (POMDP-100: 9.9, RG+LR (3s): 12, RG+LR (4s): 12). POMDP-1000 has the lowest FP across subjects at 4.8. With respect to statistical analysis, the Friedman ANOVA disclosed a main effect of the type of method on the FP scores ($Q = 15.39, p < 0.05, W = 0.57$). Nenmeyi post-hoc tests revealed statistically significant differences for POMDP-100 with RG+LR both at 3 seconds ($p < 0.05$) and 4 seconds ($p < 0.05$).

Regarding NA trials, POMDP-100 shows most of the NA trials come from S2, S5 and S7, which were discussed above on the basis or their poor decoding performance. POMDP-1000 shows the same tendency, while achieving some NA trials for the rest of the subjects.

For conclusion, both POMDP methods achieve running times between the static times of the two baselines, with POMDP-100 being slightly faster in average (3.46s) than POMDP-1000 (3.68s). The Friedman test disclosed a main effect of the method on the avg. decoding time ($Q = 26.523, p < 0.001, W = 0.98$). Nenmeyi post-hoc tests revealed significant differences between POMDP-100 ($p < 0.05$) and RG+LR at 4s, as well as between RG+LR at 3s and POMDP-1000 ($p < 0.05$). Similarly to Experiment 1, POMDP achieved a balance in accuracy and response time depending on the performance of the subject, without the need to specify a fixed trial duration for all the subjects a priori.

V. Discussion

In this study, we proposed a single POMDP model and show how it can be applied to different types of BCI paradigms (SSVEP, C-VEP and MI). The main advantage of the POMDP lies in being a flexible and integrated framework that can adapt to the underlying process that one wishes to model, and in the present study, we have observed that the POMDP model can bring distinct advantages to various BCI modalities.. This enables a sequential decision-making framework for any BCI, and allows integrating other data modalities (i.e. other

TABLE III
Experiment 3 (MI BCI) results (Test size: 72 trials)

	RG + LR (3s)			RG + LR (4s)			POMDP (cost 100)				POMDP (cost 1000)			
Sub.	Acc	FP	Avg. Time	Acc	FP	Avg. Time	Acc	FP	Avg. Time	NA	Acc	FP	Avg. Time	NA
1	0.77	10	3.0	0.81	9	4.0	0.85	7	3.21	0	0.90	4	3.44	1
2	0.56	21	3.0	0.56	21	4.0	0.46	15	3.94	11	0.0	0	4.0	48
3	0.92	4	3.0	0.92	4	4.0	0.92	4	3.16	0	0.92	3	3.38	1
4	0.71	13	3.0	0.71	13	4.0	0.65	15	3.52	2	0.63	9	3.86	9
5	0.65	17	3.0	0.71	13	4.0	0.60	9	3.74	10	0.0	0	4.0	48
6	0.60	19	3.0	0.58	20	4.0	0.60	18	3.34	1	0.56	16	3.65	5
7	0.73	13	3.0	0.63	18	4.0	0.71	10	3.85	4	0.15	1	3.94	40
8	0.96	1	3.0	1.0	0	4.0	1.0	0	3.11	0	1.0	0	3.31	0
9	0.77	10	3.0	0.79	10	4.0	0.75	11	3.23	1	0.67	10	3.55	6
Avg.	**0.74**	**12.0**	**3.0**	**0.75**	**12.0**	**4.0**	**0.73**	**9.9**	**3.46**	**3.2**	**0.53**	**4.8**	**3.68**	**17.6**
(%)	–	**17**	–	–	**17**	–	–	**14**	–	**4**	–	**7**	–	**24**

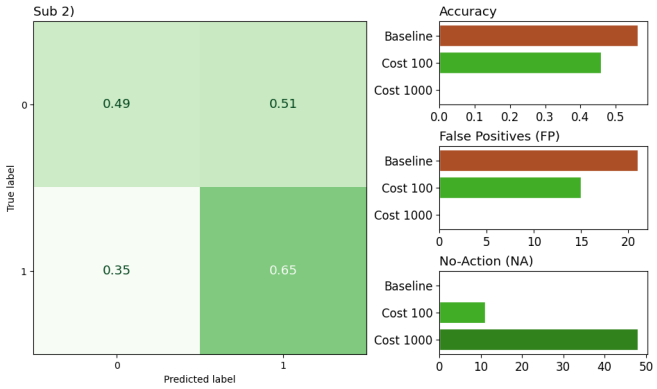

Fig. 5. Confusion matrix for S2 of Experiment 3. This confusion matrix is that obtained using the O-Cal data (Fig. 1, green) and represent the observation model of the POMDP. On the right, horizontal bar plots show performance comparison between baseline at 3s of data, POMDP-100 and POMDP-1000 for accuracy (top), false positives (mid) and no-action trials (bottom).

physiological signals) as well as other BCI pipelines, or any combination of systems that one wishes to explore.

While the POMDP model used in all three paradigms is the same, its benefits for different BCI pipelines vary depending on the characteristics of the task and the decoding algorithm being employed.

In Experiment 1, we used the POMDP model to decide on the classification of SSVEP stimuli, with the underlying reference algorithm being TRCA [11]. The classic implementation of the algorithm is static (i.e., requires a fixed epoch length to be set for every participant) and always outputs a prediction, which limits its usability for real-world scenarios. Epochs of 0.5s are considered enough to get good performance, but some participants may need more time to get good performance. The POMDP model offers a significant advantage by enabling TRCA to flexibly accumulate single predictions, achieving a delicate balance between high performance and swift decision-making. Moreover, POMDP allows for NA trials, where the model decides to skip the trial without outputing any decision if data is considered inconclusive.

Our POMDP-100 achieved a performance comparable (albeit lower) to TRCA at 1s, with an average time of 0.65s per trial. Alternatively, POMDP-1000 presents a drop in accuracy that is comparable to TRCA at 0.5s, together with a significant increase in response time in exchange for the lowest number of FP and the highest number of NA.

The most notorious disadvantage of POMDP compared to TRCA alone is the extra data needed to estimate the observation model (O-cal, Fig. 1) from the trained algorithm. In this experiment, the POMDP approach used 8 blocks for TRCA calibration (A-cal, Fig. 1 and 5 blocks as O-cal data. However, the baseline TRCA were trained with 12 trials per class (which equals to all the data mentioned above), meaning that POMDP approaches used a TRCA model training on less data compared to baseline. The results then show that POMDP performed well even if the underlying algorithm (i.e. TRCA) is trained on less data. Due to dataset size constraints, it was not possible to compare the baseline to a POMDP model where both TRCA are trained using the same amount of data.

Lastly, there exist research on flexible decision processes for TRCA [33]. Although the method presented allows for adaptable epoch length depending on the participant, it relies on calculating probability density functions every trial, and it scales with the number of classes. The paper itself analyzes the computational performance of the approach as a potential constraint and presents an optimized approach to ensure real-time performance. Besides the aforementioned computational constrains, this method is designed specifically for TRCA, while POMDP remains a general framework. However, the aforementioned approach employs data windows of increasing length until a decision is taken, while POMDP uses a sliding window of the same length for each time step. Future studies can study modifications to the POMDP definition that allow to leverage the accumulated information of an increasing data window.

In Experiment 2, we compared our POMDP model with a similar type of stimulation, C-VEP, supported by the decision pipeline EEG2Code, which shares with POMDP features such as flexible data windows, and the possibility of having NA trials. From the three experiments performed, this is the only one that includes a dedicated decision-making process. As such, this is the experiment where POMDP performs most similarly to the baseline. Although POMDP-100 tends to allow more FP trials, the differences with the chosen baseline were found to be non-significant.Meanwhile POMDP-1000 is more conservative and favors NA trials when the decoding of a subject is not good, while presenting a significant increase in avg. decoding time.

Similarly to the parameters that need to be specified for both baseline approaches (i.e. p-value and number of decisions, respectively), the cost of the POMDP needs to be decided by the experimenter. This is a constraint when observing that a higher cost is more convenient for participants whose decoding is worse, and lower cost if preferable for those performing well. Besides the generalist nature of POMDP, it faces similar constraints to EEG2Code when it comes to C-VEP decoding. Although, again, POMDP requires extra data to estimate the observation model, and the division of the data was identical as that of Experiment 1 both for the POMDP models and for the baseline. This means that the C-VEP baseline analyses were performed including said extra data in the training of the CNN. The results equally show that POMDP can perform comparatively to EEG2Code with less calibration data. Future studies could implement modifications to the POMDP model such that it can adapt the cost depending on decoding performance, since the performance of the decoding algorithm is required to estimate the observation model.

As previously mentioned when discussing the results of Experiment 1, the current POMDP model uses sliding data windows of fixed length in each step, while both of the baseline methods use increasing data windows in each step. If the POMDP model could leverage the existing data to get more information in each time step, it seems natural to assume that overall performance would increase. Because of that, it is a prominent future direction for the POMDP-BCI framework. This, together with the fact that both approaches employ

advanced decision-making processes, could explain why this is the only experiment where POMDP tends to produce more FP trials when compared to the baseline.

Experiment 3 analyzed the performance of the proposed POMDP model for a 2-class MI BCI. As with Experiment 1, the baseline used a static time window, that was set to 3 and 4 seconds, respectively. On the one hand, POMDP-100 achieves a significant reduction of the average time with respect to the baseline at 4 seconds, while being adapted to the decoding performance of each participant. On the other hand, POMDP-1000 achieves a significant reduction of FP trials with respect to both baselines, while introducing a significant increase in decoding time when compared with the baseline at 3 seconds.

Interestingly, the POMDP-1000 model sees a dramatic decrease in accuracy performance. We further investigated and found that the general accuracy drop was mostly due to three subjects' accuracy (that of S2, S5 and S7) decreasing either to or near 0 for POMDP-1000. Upon inspecting the confusion matrices of these three subjects (Fig. 5), which are used as the observation model for their respective POMDPs, we found that the aforementioned subjects exhibit near-chance classification performance for at least one of the classes. This makes the POMDP decide not to act and results on a high number of NA trials. MI BCIs are known to require training in order to perform well [41]. This, together with the phenomenon known as 'BCI illiteracy' (i.e., the fact that some users seem to be unable to achieve a good performance in MI BCI systems) [42], can explain that for some subjects the approaches perform poorly. The POMDP model at high cost can adapt to this particularity and decide to not act when the user's performance in the MI task is poor. This behavior could be particularly interesting for applications where FP trials represent a threat to the user, and it is preferable to not do anything if the user faces difficulties performing the task.

Contrary to the previous two experiments, the baseline algorithm does not include the data that corresponded to the estimation of the observation model of the POMDP approaches. This may lead the reader to suggest the comparison between POMDP and baseline is not fair in this experiment. However, including this information in the baseline algorithm would introduce data from the second session, invalidating the comparison with the POMDP model, whose decoding algorithm is trained only with data from the first session. For the sake of making comparisons between two models that perform inter-session classification, we decided to exclude this data from the baseline comparison. Since the observation model represents the probability of observing different classification outputs given the observed data, it only makes sense to estimate the observation model using data from the same session as the testing data for inter-session classification.

In a real-life situation, one can think as the first session being used to train the underlying decoding algorithm, and while the POMDP requires more data to estimate the observation model, nothing stops the user to start the second session using the baseline algorithm alone. The data from the initial trials can be stored and, once enough data are available, the POMDP can be solved and deployed to replace the baseline algorithm.

Future studies can explore this scenario, as well as dive into the potential applications of POMDP for MI BCI systems that rely on continuous feedback, instead of classification outputs. Since the POMDP keeps a probability distribution over all possible state, it could potentially be explored as a means of providing a certainty score for each class that updates every time step.

Similarly to experiment 1, we observed how the cost of the POMDP determined performance in terms of the speed/accuracy trade-off: while both costs achieve adaptive decoding times, POMDP-100 tends to allow more FP in favor of faster decoding times, while POMDP-1000 prioritizes reducing FP trials at the cost of slower decoding times. As previously mentioned when discussing Experiment 2's results, the fact that the cost needs to be decided beforehand becomes a relevant shortcoming for POMDP applied to BCI in the light of this work's results. As such, a promising future work for POMDP would be to design it such as the cost can be automatically assigned depending on the desired speed/accuracy trade-off.

VI. Conclusion

The present work shows how a POMDP-based approach can provide an integrated sequential decision framework to those BCI approaches that lack it on their classic implementations, while performing comparatively to those that already include it. Moreover, the fact that the same model can easily be adapted to every modality means that combination of BCI with other data modalities (e.g. other physiological measurements or behavioral data) or other BCI pipelines would be possible within the same POMDP-based framework.

Future work can explore such an integration, as well as potential modifications to the POMDP model definition that address the shortcomings found in the present study, namely the constraint of the POMDP to use (sliding) fixed-length data windows for each time step, as well as the need to specify the cost beforehand. The former could be achieved by including a *time step* dimension in the observation model, having a different observation model for each time step that reflects decoding performance with the corresponding amount of data, while the latter could be achieved by specifying different costs depending on the estimated observation model. This could be achieved by mapping decoding performance to a cost via some uncertainty metric, similar to the ρ-POMDP [43] model.

References

[1] J. R. Wolpaw, N. Birbaumer, W. J. Heetderks, D. J. McFarland, P. H. Peckham, G. Schalk, E. Donchin, L. A. Quatrano, C. J. Robinson, T. M. Vaughan *et al.*, "Brain-computer interface technology: a review of the first international meeting," *IEEE transactions on rehabilitation engineering*, vol. 8, no. 2, pp. 164–173, 2000.

[2] P. Wierzgała, D. Zapała, G. M. Wojcik, and J. Masiak, "Most popular signal processing methods in motor-imagery bci: a review and meta-analysis," *Frontiers in neuroinformatics*, vol. 12, p. 78, 2018.

[3] B. Liu, X. Chen, N. Shi, Y. Wang, S. Gao, and X. Gao, "Improving the performance of individually calibrated ssvep-bci by task-discriminant component analysis," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 1998–2007, 2021.

[4] I. A. Fouad, F. E.-Z. M. Labib, M. S. Mabrouk, A. A. Sharawy, and A. Y. Sayed, "Improving the performance of p300 bci system using different methods," *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 9, pp. 1–13, 2020.

[5] P. F. Diez, A. G. Correa, L. Orosco, E. Laciar, and V. Mut, "Attention-level transitory response: a novel hybrid bci approach," *Journal of neural engineering*, vol. 12, no. 5, p. 056007, 2015.

[6] G. Pfurtscheller, T. Solis-Escalante, R. Ortner, P. Linortner, and G. R. Muller-Putz, "Self-paced operation of an ssvep-based orthosis with and without an imagery-based "brain switch:" a feasibility study towards a hybrid bci," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 18, no. 4, pp. 409–414, 2010.

[7] Y. Li, J. Pan, F. Wang, and Z. Yu, "A hybrid bci system combining p300 and ssvep and its application to wheelchair control," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 11, pp. 3156–3166, 2013.

[8] L. Cao, J. Li, H. Ji, and C. Jiang, "A hybrid brain computer interface system based on the neurophysiological protocol and brain-actuated switch for wheelchair control," *Journal of neuroscience methods*, vol. 229, pp. 33–43, 2014.

[9] T. Yu, J. Xiao, F. Wang, R. Zhang, Z. Gu, A. Cichocki, and Y. Li, "Enhanced motor imagery training using a hybrid bci with feedback," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 7, pp. 1706–1717, 2015.

[10] M. Nakanishi, Y. Wang, Y.-T. Wang, and T.-P. Jung, "A comparison study of canonical correlation analysis based methods for detecting steady-state visual evoked potentials," *PloS one*, vol. 10, no. 10, p. e0140703, 2015.

[11] M. Nakanishi, Y. Wang, X. Chen, Y.-T. Wang, X. Gao, and T.-P. Jung, "Enhancing detection of ssveps for a high-speed brain speller using task-related component analysis," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 1, pp. 104–112, 2017.

[12] X. Chen, Z. Chen, S. Gao, and X. Gao, "A high-itr ssvep-based bci speller," *Brain-Computer Interfaces*, vol. 1, no. 3-4, pp. 181–191, 2014.

[13] X. Chen, Y. Wang, S. Gao, T.-P. Jung, and X. Gao, "Filter bank canonical correlation analysis for implementing a high-speed ssvep-based brain–computer interface," *Journal of neural engineering*, vol. 12, no. 4, p. 046008, 2015.

[14] X. Zhang, L. Yao, X. Wang, J. Monaghan, D. Mcalpine, and Y. Zhang, "A survey on deep learning based brain computer interface: Recent advances and new frontiers," *arXiv preprint arXiv:1905.04149*, vol. 66, 2019.

[15] A.-D. Mezei, L. Tamás, and L. Buşoniu, "Sorting objects from a conveyor belt using active perception with a pomdp model," in *2019 18th European Control Conference (ECC)*. IEEE, 2019, pp. 2466–2471.

[16] Y. Satsangi, S. Lim, S. Whiteson, F. Oliehoek, and M. White, "Maximizing information gain in partially observable environments via prediction reward," *arXiv preprint arXiv:2005.04912*, 2020.

[17] R. D. Smallwood and E. J. Sondik, "The optimal control of partially observable markov processes over a finite horizon," *Operations research*, vol. 21, no. 5, pp. 1071–1088, 1973.

[18] M. Ghasemi and U. Topcu, "Online active perception for partially observable markov decision processes with limited budget," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 6169–6174.

[19] M. T. Spaan, "Cooperative active perception using pomdps," in *AAAI 2008 workshop on advancements in POMDP solvers*, 2008.

[20] M. Spaan and P. Lima, "A decision-theoretic approach to dynamic sensor selection in camera networks," in *Proceedings of the International Conference on Automated Planning and Scheduling*, vol. 19, 2009, pp. 297–304.

[21] P. Natarajan, P. K. Atrey, and M. Kankanhalli, "Multi-camera coordination and control in surveillance systems: A survey," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 11, no. 4, pp. 1–30, 2015.

[22] A. Guo, "Decision-theoretic active sensing for autonomous agents," in *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, 2003, pp. 1002–1003.

[23] M. Araya, O. Buffet, V. Thomas, and F. Charpillet, "A pomdp extension with belief-dependent rewards," *Advances in neural information processing systems*, vol. 23, 2010.

[24] M. T. Spaan, T. S. Veiga, and P. U. Lima, "Decision-theoretic planning under uncertainty with information rewards for active cooperative perception," *Autonomous Agents and Multi-Agent Systems*, vol. 29, no. 6, pp. 1157–1185, 2015.

[25] T. Gateau, C. P. C. Chanel, M.-H. Le, and F. Dehais, "Considering human's non-deterministic behavior and his availability state when designing a collaborative human-robots system," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4391–4397.

[26] P. E. U. de Souza, C. P. C. Chanel, and F. Dehais, "Momdp-based target search mission taking into account the human operator's cognitive state," in *2015 IEEE 27th international conference on tools with artificial intelligence (ICTAI)*. IEEE, 2015, pp. 729–736.

[27] G. Singh, R. N. Roy, and C. P. Chanel, "Pomdp-based adaptive interaction through physiological computing," in *HHAI2022: Augmenting Human Intellect*. IOS Press, 2022, pp. 32–45.

[28] M. J. Bryan, S. A. Martin, W. Cheung, and R. P. Rao, "Probabilistic co-adaptive brain–computer interfacing," *Journal of neural engineering*, vol. 10, no. 6, p. 066008, 2013.

[29] J. Park, K.-E. Kim, and S. Jo, "A pomdp approach to p300-based brain-computer interfaces," in *Proceedings of the 15th international conference on Intelligent user interfaces*, 2010, pp. 1–10.

[30] X. Chen, Y. Wang, M. Nakanishi, X. Gao, T.-P. Jung, and S. Gao, "High-speed spelling with a noninvasive brain–computer interface," *Proceedings of the national academy of sciences*, vol. 112, no. 44, pp. E6058–E6067, 2015.

[31] Y. Wang, Y. Wang, and T.-P. Jung, "Visual stimulus design for high-rate ssvep bci," *Electronics letters*, vol. 46, no. 15, p. 1, 2010.

[32] H. Kurniawati, D. Hsu, and W. Lee, "SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces," in *Proc. RSS*, 2008.

[33] J. Tang, M. Xu, J. Han, M. Liu, T. Dai, S. Chen, and D. Ming, "Optimizing ssvep-based bci system towards practical high-speed spelling," *Sensors*, vol. 20, no. 15, p. 4186, 2020.

[34] V. Jayaram and A. Barachant, "Moabb: trustworthy algorithm benchmarking for bcis," *Journal of neural engineering*, vol. 15, no. 6, p. 066011, 2018.

[35] K. Zheng and S. Tellex, "pomdp_py: A framework to build and solve pomdp problems," in *ICAPS 2020 Workshop on Planning and Robotics (PlanRob)*, 2020, arxiv link: "https://arxiv.org/pdf/2004.10099.pdf". [Online]. Available: https://icaps20subpages.icaps-conference.org/wp-content/uploads/2020/10/14-PlanRob_2020_paper_3.pdf

[36] L. Darmet, S. Ladouce, and F. Dehais, "Shortened calibration of code-vep based bci by improved deep learning architecture and golden subjects pre-training."

[37] S. Nagel and M. Spüler, "World's fastest brain-computer interface: combining eeg2code with deep learning," *PloS one*, vol. 14, no. 9, p. e0221909, 2019.

[38] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K. J. Miller, G. Mueller-Putz *et al.*, "Review of the bci competition iv," *Frontiers in neuroscience*, p. 55, 2012.

[39] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, "Multiclass brain–computer interface classification by riemannian geometry," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 4, pp. 920–928, 2011.

[40] ——, "Classification of covariance matrices using a riemannian-based kernel for bci applications," *Neurocomputing*, vol. 112, pp. 172–178, 2013.

[41] C. Neuper and G. Pfurtscheller, "Neurofeedback training for bci control," *Brain-Computer Interfaces: Revolutionizing Human-Computer Interaction*, pp. 65–78, 2010.

[42] B. Blankertz, C. Sanelli, S. Halder, E. Hammer, A. Kübler, K.-R. Müller, G. Curio, and T. Dickhaus, "Predicting bci performance to study bci illiteracy," *BMC Neurosci*, vol. 10, no. Suppl 1, p. P84, 2009.

[43] V. Thomas, G. Hutin, and O. Buffet, "Monte carlo information-oriented planning," *arXiv preprint arXiv:2103.11345*, 2021.