A Self-Adaptive Online Brain Machine Interface of a Humanoid Robot through a General Type-2 Fuzzy Inference System

Javier Andreu-Perez, Fan Cao, Hani Hagras, and Guang-Zhong Yang

Abstract—This paper presents a self-adaptive general type-2 fuzzy inference system (GT2 FIS) for online motor imagery (MI) decoding to build a brain-machine interface (BMI) and navigate a bi-pedal humanoid robot in a real experiment, using EEG brain recordings only. GT2 FISs are applied to BMI for the first time in this study. We also account for several constraints commonly associated with BMI in real practice: 1) maximum number of electroencephalography (EEG) channels is limited and fixed, 2) no possibility of performing repeated user training sessions, and 3) desirable use of unsupervised and low complexity features extraction methods. The novel learning method presented in this paper consists of a self-adaptive GT2 FIS that can both incrementally update its parameters and evolve (a.k.a. self-adapt) its structure via creation, fusion and scaling of the fuzzy system rules in an online BMI experiment with a real robot. The structure identification is based on an online GT2 Gath-Geva algorithm where every MI decoding class can be represented by multiple fuzzy rules (models). The effectiveness of the proposed method is demonstrated in a detailed BMI experiment where 15 untrained users were able to accurately interface with a humanoid robot, in a single thirty-minute experiment, using signals from six EEG electrodes only.

Index Terms—General type-2 fuzzy systems, online brain machine interfaces, motor-imagery brain machine interfaces, self-adaptive learning, adaptive learning, phase synchrony features.

I. INTRODUCTION

THE application of brain-machine interfaces (BMIs) methods to the control of external devices and robots based on neural activity has raised increasing interests in recent years. Non-invasive modalities such as electroencephalography (EEG) have been widely adopted due to their ease of use and lower cost compared to other brain monitoring techniques allowing their use for real world applications [1, 2]. Yet, the effective development of such systems for wide acceptability is faced with many challenges. In particular, an important issue is concerned with poor signal-to-noise ratio and high non-stationarity [3, 4]. This is particularly the case when users are first initiated to BMI and more prone to exhibiting a non-stationary signal, with some even unable to produce stable EEG patterns. This problem is further intensified when BMI users are directly introduced to a real robot control scenario without prior experience.

The goal of motor imagery BMI (MI-BMI) is to find a pattern in the EEG signal underlying the cognitive process associated with thinking about a specific motor command that implies movement [5]. In most cases, MI-BMI indeed requires users to go through repeated sessions of the same experiment, typically over different days, in order for them to generate stable patterns and achieve the required concentration levels for reliable control. Non-stationarity is common as users tend to vary their cognitive response, when changing the context of execution (e.g. from a simple visual experiment on a computer screen, to the real control of an object or robot). Participants can therefore become frustrated [6, 7] and opt out of the experiment or require long periods (e.g. several days) of repetitive training [8].

The above challenges call for the development of effective BMIs through adaptive and online learning [9, 10]. Previous research has suggested the use of a co-learning paradigm as a solution to the above issues [11]. This paradigm is based on the extension of an initial basic session with an online adaptive BMI learning stage in which the user's experience and the machine learning algorithm are jointly improved. Computationally expensive and supervised features are not recommended since their supervised nature adds another subjacent source of decalibration due to non-stationarity and their computational demand can indeed compromise the requirement of performing the learning online as proposed.

Fuzzy logic allows handling the uncertainty associated with the noisy signals encountered in realistic real-time scenarios and BMI applications. Indeed, general type-2 (GT2) fuzzy rules have been shown to provide higher robustness to noisy data and unexpected signal patterns such as extreme values and outliers [9, 10]. As EEG signals are non-stationary and the user's brain behavior is highly unpredictable, general type-2 fuzzy inference systems (GT2 FISs) offer an excellent framework for learning using multiple models (i.e. rules in this case) in this remarkably noisy and changing BMI context. At the time when this article was written, we were not aware of a previous implementation of GT2 FISs in BMI. Additionally, rather than just tuning the model parameters, we propose a learning mechanism for GT2 FISs that provides a higher level of adaptation enabling the online creation of new models, merging or re-scaling them in order to adapt to the unpredictable nature and unknown of brain activity.

In this paper, we present a GT2 fuzzy logic online self-adaptive Gath-Geva method, with the following novelties: 1) first implementation of GT2 FISs in BMI; 2) a novel self-adaptive learning algorithm based on GT2 FISs with a system structure that is not fixed a priori but can self-develop as well as adapt its parameters across the course of the online scenario.

Our methodology is evaluated with the realistic navigation of a bipedal robot through the sole use of MI commands. The users are presented with an audio neuro-feedback with either

increasing or decreasing volume given the online fuzzy confidence of the predicted MI class.

In a nutshell, for the proposed experiment we set the following objectives and constraints: 1) Fast training of a group of participants for the control of a humanoid bipedal robot through MI; 2) Achieving high-accuracy MI robot control with as few trials as possible and no repeated sessions; 3) Only a small set of EEG measurement electrodes must be used and their number and locations should remain identical for all participants; 4) The features extracted from the EEG signals must be simple and few so as to permit fast real-time processing and model adaptation.

After an overview of the related work in Section II, Section III presents the proposed self-adaptive GT2 fuzzy classification algorithm. Then, Section IV describes online self-adaptive brain machine interface experiment. Finally, Section V presents the experiment and results, followed by the conclusions and future work in Section VI.

II. RELATED WORK

This section introduces the related work that served as background and motivation for the present research:

A. Motor Imagery BMI

MI is a paradigm of particular interest to BMI research. In Millan et. al. [11], eight fronto-centro-parietal EEG channels were used to build a MI-BMI system based on two mental tasks (left and right movements). The processing method involved transforming the EEG potentials by means of surface Laplacian and a local neural classifier, yielding an overall accuracy of 70%. Despite this, the classification model was not always capable of responding consistently, as reflected by an accuracy of only 20-25% for some users. Further work from the same group reported greater performance stability across all volunteers when considering a larger number of channels (up to 26) [11, 12], although repeated training sessions over three to five days were required for acceptable performance levels. In the work by LaFleur et al. [13], BMI was used for controlling a quadcopter via MI decoding of three movements (i.e. left hand, right hand and both hands), the users were able to control the device with overall accuracies ranging from 70 to 90% by the end of the study. Such results involved the use of a high-density EEG array of 64 channels as well as three days to three months of user training.

The Berlin BMI group provides a collection of datasets organized by competitions that took place in different years. In Competition III, considering experiments of more than two classes, the smallest number of electrodes was 32 while the highest was 118, applying a processing methodology based on the subspace transformation of the EEG signals with offline classifiers [14]. This number in turn varied from 22 to 64 in Competition IV, where the best method used a supervised filter algorithm and common spatial pattern (CSP) combined with a set of offline classifiers [15]. Before testing, both experiments also required participant-specific, offline optimization searches, including the selection of relevant channels from the whole set of classifier parameters in order to improve accuracy.

It is worth mentioning that the number of EEG electrodes used in this competition was much larger than the one proposed in the present study. Moreover, as regards the experimental setting, the mental task suggested on a screen had to be performed through simulated control; however, a translation into realistic BMI control was lacking. Yet, when BMI users are asked to simultaneously control a device, their cognitive perception is altered by the transfer of the experimental context and the attentional demands of the experiment, which can in turn alter their way of planning and executing the MI. In addition, most recruited participants were not defined as being new to BMI.

B. Adaptive Motor Imagery BMI

As a replacement for the brain's normal neuromuscular channels, BMI systems crucially rely on feedback and should be able to adapt to accommodate the changes in the user's brain behaviors as a response to this feedback. This paradigm was first envisioned by Wolpaw et al. [16] and later coined as co-adaptive BMI by Vidaurre et. al. [17] who proposed two highly referenced algorithms, namely Adaptive Information Matrix (AdIM) and an online Linear Discriminant Analysis (oLDA) based on Kalman updates. In [18] the oLDA method was extended to allow unsupervised parameter updates of the weights and bias of the linear discriminant using the pooled mean and global sample covariance. This method has been employed with satisfactory performance in several other works. oLDA has been mainly proposed in conjunction with supervised feature extraction methods (CSP) and its model structure is fixed.

C. Fuzzy Inference Systems and Clustering in BMI

Fuzzy inference provides a framework of particular interests to cope with the uncertainties typically associated with BMI applications, as follows: 1) uncertainties related to the reliability and quality of the electrode measurements as a results of evolving conditions across the experiment; 2) uncertainty about the user's cognitive profile, i.e. users may alter the cognitive profile intentionally or unintentionally during the experiment [19]; 3) changes in the environmental conditions; 4) considering the BMI context, users may have different perceptual demands for different tasks, hence the uncertainties as to whether the same cognitive patterns will replicate for the same intentions. The first three are practically inherent to any BMI application, while the fourth one is raised by the goal of obtaining real-life BMI systems, where task-specific training data is difficult to obtain or very limited.

Type-2 fuzzy systems have been marginally applied to BMI with varying results. In the pioneer work by Herman et. al. [20, 21], interval type-2 (IT2) fuzzy logic based systems outperformed conventional methods applied to BMI such as SVMs and LDA, as well as classic type-1 fuzzy inference systems (T1 FISs). Das et al. [22] proposed an IT2 fuzzy inference system (IT2 FIS) with an online adaptable and self-adaptive structure for MI BMI. Nevertheless, the extracted EEG features were based on CSP [23] and its logic is still based on interval fuzzy sets.

An unsupervised fuzzy clustering algorithm known as Gath-Geva (GG) was originally designed for the detection of sleeping

stages from EEG data [24]. As a difference to fuzzy c-means (FCM), GG is an efficient algorithm for modelling arbitrary shaped clusters based on elliptical shapes defined by the general structure of covariance matrices. Besides, it exerts a level of control over the sensitivity of the number of points for each cluster, which is lacking in other fuzzy clustering algorithms such as FCM and Gustafson-Kessel (GK) [25]. GG algorithm uses a combination of cluster volume, fuzzy membership and frequentist inference in order to provide a flexible yet constrained adaptation, which is suitable for noisy, artifact-prone and non-stationary signals, as with EEG. Additionally, the use of GT2 fuzzy sets provides an extra layer of protection towards the consolidation of stable models [9, 10]. Samples laying on the lower cuts in the third dimension weigh little during the inference process, while inversely intense areas around the cluster prototypes receive a large weight, thus consolidating the rule. This enhanced management of noise and extreme values positively contributes to improving performance with respect to current state-of-the-art methods.

III. Self-Adaptive General TYPE-2 FUZZY CLASSIFICATION ALGORITHM

The proposed novel approach relies on an online unsupervised fuzzy GG algorithm to discover the prototypes (rule antecedents). Each prototype describes a relevant pattern in the EEG features. As the rules are fuzzy, each prototype can be associated to several classes and vice-versa. In order to associate each rule with the target BMI commands, an initial degree of association of each prototype (consequent) with a determined class is set by the programmer intuition or by processing a small labelled sample from a previous MI source task. In this paper, we use a small sample set from a standard offline screen-based task (basic task) prior to interfacing with the robot. During the online robot operation, as no more labelled samples can be provided, it is the responsibility of the learning algorithm to keep self-adapting to accommodate to the new and non-deterministic behavior of the incoming EEG feature data in real-time. Self-adaptation includes new generation, fusion and scaling of the model rules, as well as numerical adaptation of their model parameters. This section provides the background and detailed explanation of the proposed learning algorithm

A. Type-1 Fuzzy GG Algorithm

The original fuzzy GG algorithm tries to seek an exhaustive segmentation of the data that employs the gradual fuzzy membership of each point to a set of clusters. Let us define a dataset described by $X = \{x_1, x_2, x_3, ..., x_N\}$, $X \in R^n$ where n is the dimensionality of the dataset. The idea behind every fuzzy clustering algorithm is to minimize the following objective function:

$$J = \sum_{j=1}^{R} \sum_{i=1}^{N} u_{i,j}^m \, d_{i,j} \,, \qquad (1)$$

where $j \in (1, 2, ..., R)$ is the number of clusters, and $u_{i,j}$ is the membership degree of the ith sample data with respect to the jth cluster. An important element in this equation is m, i.e. the exponent of the membership function. This exponent defines the defuzzification degree, also called "fuzzifier" parameter.

This value is relevant to the location and quality of the cluster. It identifies the amount of uncertainty that a cluster can cope with. For instance, if the value of m is close to 1, then the algorithm results in crisp clusters while if the value is greater than 3, then it results in spiky clusters. A common choice for the value of the parameter m is 2 [26, 27]. However, it has been noted in biological experiments that a value of 2 is not always the most suitable [28, 29]. Adding fuzziness on the parameter m provides a new dimension of uncertainty, which is a key factor for the extension of the fuzzy clustering algorithm from type-1 to type-2 fuzzy logic. As the distance of a same sample to the jth rule center is denoted as $d_{i,j}$ and GG algorithms use a likelihood function to define distances, the distance is thus defined with the following equation [24]:

$$d_{ij} = \frac{|\Sigma_j|}{p_j} \exp\left(-\frac{1}{2}(x_i - v_j)^T \Sigma_j^{-1}(x_i - v_j)\right), \qquad (2)$$

which is based on a maximum likelihood estimation. The parameter v_j denotes the center of rule with Σ_j being its corresponding covariance matrix. The symbol $|...|$ represents the determinant of the rule covariance while P_j is a measure indicating the *a priori* probability of the degree of association of a sample with a rule. This value is computed as the expectation from all the rule membership values that have been observed so far, which can be estimated with:

$$p_j = \frac{1}{N} \sum_{i=1}^{N} u_{i,j} \,, \qquad (3)$$

This value serves for eliminating the sensitivity of the algorithm to a set of clusters with an imbalanced number of points. The membership $u_{i,j}$ of the value that makes the cluster partition function can be computed as:

$$u_{i,j} = \frac{1}{\sum_{q=1}^{R} (d_{i,j} \,/\, d_{i,q})^{2/(m-1)}} \,. \qquad (4)$$

Following this, we now have all the elements enabling to compute the centroid and covariance of the cluster with the below expressions:

$$v_j = \sum_{i=1}^{N} x_i u_{i,j}^m \Big/ \sum_{i=1}^{N} u_{i,j}^m \,, \qquad (5)$$

$$\Sigma_j = \sum_{i=1}^{N} u_{i,j}^m (x_i - v_j)(x_i - v_j)^T \Big/ \sum_{i=1}^{N} u_{i,j}^m \,, \qquad (6)$$

The whole algorithm is iterated until a stopping condition is accomplished, such as:

$$\max_{i,j} \left| u_{i,j}^{iter} - u_{i,j}^{iter-1} \right| < \varepsilon \,, \qquad (7)$$

where $iter$ indicates the index of the iteration and ε is a termination criterion usually set to a value between 0 and 1. The pseudocode of the type-1 fuzzy GG algorithm is detailed in Table I.

TABLE I: TYPE-1 GG FUZZY ALGORITHM

	Input:	Vector x, trial labels l, fuzziness level $m \in \{1, \infty\}$, number of clusters R.
	Output:	Clusters v, Σ and u.
1	*Repeat*	
2		*For* i=1 to *N*
3		Calculate distance from each center to all data samples by Eq. (2).
4		Compute degree of membership by the fuzzy partition function for all data samples by Eq. (4).

5		Estimate fuzzy center of each cluster by Eq. (5).
6		Estimate fuzzy covariance of each cluster by Eq. (6).
7	*End*	
8	*If* condition (7) is satisfied	
9		Break loop.
10	*end*	
11	*end*	

B. Offline Type-1 Fuzzy GG Classifier

Unsupervised GG fuzzy algorithm can be converted into a supervised classifier by slightly extending its formulation to convert the set of clusters (a.k.a. rules) into a fuzzy mixture model [30]. Hence, the consequences are defined by a rule class association, denoted as $h_{j,k}$ multiplied by the firing of a rule, expressed as $f_{i,j}$. The rule base is then defined in the following way:

$$r_j: \quad \begin{matrix} \text{IF } x_i \text{ is around } v_j \text{ then } y_j = k \text{ with } h_{j,k} \cdot f_{i,j}, \\ \ldots, y_j = K \text{ with } h_{j,K} \cdot f_{i,j} . \end{matrix} \quad (8)$$

The firing level of each rule is calculated as the quadratic multivariate classifier term [30]:

$$f_{i,j} = \frac{p_j}{|\Sigma_j|} \exp\left(-\frac{1}{2}(x_i - v_j)^T \Sigma_j^{-1}(x_i - v_j)\right), \quad (9)$$

which is analogous to the distance, with the difference that the first term of the exponential value is inverted. The second term, in this case, is the degree of association of a sample with a rule. As a difference to standard fuzzy rule-based (FRB) systems, since the firing level of a rule is already multivariate as in (9), a t-norm, which would have been represented by the logical operator "AND" over the feature space, is not used here. The rule class association is defined as the ratio of the sum of memberships for each class:

TABLE II: TYPE-1 FUZZY GG CLASSIFIER

	Input:	Vector x, trial labels l, fuzziness level $m \in \{1, \infty\}$, initial fuzzy membership u, initial number of rules K.
	Output:	Rules v, Σ, u, p_j, f_j and $h_{j,k}$.
1	Repeat	
2		Estimate the fuzzy center of each rule by Eq. (5).
3		Estimate the fuzzy covariance of each rule by Eq. (6).
4		Compute degree of membership by the fuzzy partition function for all data samples by Eq. (4).
5		Estimate rule class associations by Eq. (10).
6		Calculate a priori probability of the degree of association by Eq. (3)
7		Compute distance from all samples by Eq. (11).
8		Estimate fuzzy partition memberships by Eq. (4).
9		Compute degree of activation for each rule by Eq. (9)
10		If condition (7) is satisfied
11		Break loop.
12		end
13	end	

$$h_{j,k} = \frac{\sum_{i=1}^{N} u_{i,j}^m \, \delta(k, l_i)}{\sum_{i=1}^{N} u_{i,j}^m}, \quad \delta(k, l_i) \begin{cases} k = l_i, & 1, \\ k \neq l_i, & 0, \end{cases} \quad (10)$$

where δ is a Kronecker delta function that takes a class index k_i indicating a specific class and l_i as the class label of the *i*th sample. Likewise, the original distance formulated by GG does not provide any measure of affiliation to a particular MI class. In order to provide the rules with an initial supervision on the

target BMI commands, either a small set of trials or current predictions can be used. Then the distance measurement in (2) is redefined as:

$$d_{i,j} = h_{j,k=l_i} \frac{|\Sigma_j|}{p_j} \exp\left(-\frac{1}{2}(x_i - v_j)^T \Sigma_j^{-1}(x_i - v_j)\right). \quad (11)$$

The final class of the sample is computed by the weighted average of the rule outputs, as center-of-gravity defuzzification:

$$y_{i,k} = \frac{\sum_{j=1}^{R} f_{i,j} h_{j,k}}{\sum_{j=1}^{R} f_{i,j}}, \quad k^* = \arg\max_{1 \le k \le K} y_{i,k}, \quad (12)$$

where K is the number of classes and k^* the resulting predicted class.

For a detailed background on the similarities of the method with a Gaussian Mixtures of Bayes classifier, we refer the reader to the work by Abony and Szeifert [30]. The pseudocode of the type-1 fuzzy GG classifier is described in Table II.

C. General Type-2 Fuzzy Gath-Geva (GT2FGG) Classifier

From the formulation of the type-1 supervised model described in the previous section, we propose a set of amendments to extend the approach presented in the previous section to GT2 fuzzy. To this end, we adopt the method proposed by [10], where the cluster fuzzifier *m* is represented as a fuzzy set denoted by M. As the computation of the primary membership function $u_{i,j}$ is dependent on the value of *m*, we can understand this fuzzy set M as the fuzzy membership grades of $u_{i,j}$. Knowing that we have a primary and secondary inter-related membership functions, we can define the fuzzy membership of a point x by a type-2 fuzzy membership function $\mu_{\tilde{A}}(x,u)$, representing the uncertainty related to the input data as a GT2 fuzzy set, itself defined as [9], [31]:

$$\tilde{A} = \{((x,u), \tilde{\mu}(x,u)) \mid \forall_x \in X \; \forall u \in J_x \subseteq [0,1]\}, \quad (13)$$

or in the equivalent form:

$$\int_{x \in X} \int_{u \in J_x} \tilde{\mu}(x,u) / (x,u), \quad J_x \subseteq [0,1], \quad (14)$$

where the integral means the union of all cases of x and u [31]. Early representations of GT2 were computationally expensive [31] and less applicable to the fast input-output response required for BMI scenarios. Fortunately, alternative representations of GT2 have been introduced enabling to use them with efficient computation as in the work reported in [32] and [33]. Both [32] and [33] aim at representing GT2 as a composition of multiple IT2 fuzzy sets. The relation of both methods in their terminology, which is explained in [32].

In the present paper we will use the notation of *zSlices*. A *zSlice* is formed by cutting a plane at level z_e of the GT2 fuzzy sets over what is denoted as the third dimension that spans over the secondary membership. This slicing results in a zSlice $\tilde{Z}_e$, which is defined as follows [33]:

$$\tilde{Z}_e = \int_{x \in X} \int_{u_e \in J_{e_x}} z_e / (x, u_e), \quad (15)$$

or similarly [33]

$$\tilde{Z}_e = \{(x, u_e), z_e) \mid \forall_x \in X, \; \forall u_e \in J_x\}. \quad (16)$$

The GT2 fuzzy set can be recovered by integrating the collection of an infinite number of zSlices [33]:

$$\tilde{A} = \int_{0 \le e \le I} \tilde{Z}_e \quad I \to \infty, \quad (17)$$
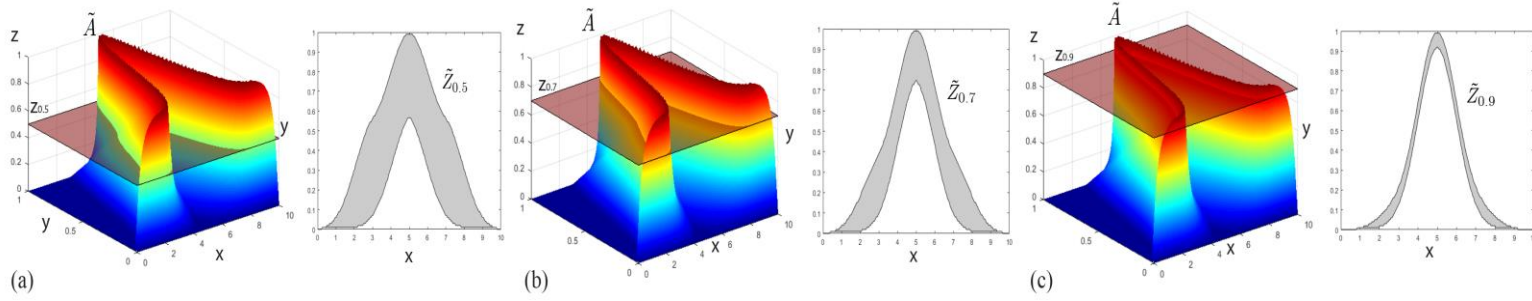
or, for a discrete number [33]:

Fig. 1. The three colored graphs in a), b), and c) display an example of a general fuzzy set $\tilde{A}$ with a cutting plane on the zLevels 0.5, 0.7, and 0.9, respectively. Their resulting zSlices are depicted in the grey-scaled graphs.

$$\tilde{A} = \sum_{e=0}^{I} \tilde{Z}_e . \qquad (18)$$

In this work, the concept of zSlices based GT2 fuzzy is reinterpreted in order to fit it to our fuzzy partition model. The zSlices-based GT2 fuzzy membership for the sample set X can be obtained as:

$$\tilde{Z}_{e,j} = \sum_{x \in X} \sum_{u \in [\bar{u}, \underline{u}]} z_e / \left[\bar{u}_{i,j}, \underline{u}_{i,j} \right] , \qquad (19)$$

where $\bar{u}_{i,j}$ and $\underline{u}_{i,j}$ are the upper and lower rule memberships, respectively for a determined zLevel. Fig. 1 shows a GT2 fuzzy set and displays some of their zSlices for selected zLevels. These memberships can be derived from the rule partition matrix with the following expressions:

$$\bar{u}_{i,j} = \max\left(\frac{1}{\sum_{q=1}^{R} (d_{i,j} / d_{i,s})^{2/(\bar{M}_{z_e}-1)}}, \frac{1}{\sum_{q=1}^{R} (d_{i,j} / d_{i,q})^{2/(\underline{M}_{z_e}-1)}} \right), \qquad (20)$$

$$\underline{u}_{i,j} = \min\left(\frac{1}{\sum_{q=1}^{R} (d_{i,j} / d_{i,s})^{2/(\bar{M}_{z_e}-1)}}, \frac{1}{\sum_{q=1}^{R} (d_{i,j} / d_{i,s})^{2/(\underline{M}_{z_e}-1)}} \right), \qquad (21)$$

where $[\underline{M}_{z_e}, \bar{M}_{z_e}]$ are the left and right bounds of the fuzzy set M with cut at z_e. The centroid of zSlices-based GT2 fuzzy sets can be computed by the composition of the centroids of its zSlices $\tilde{Z}_e$. As such, this can be defined as [33]:

$$C_{\tilde{Z}} = \sum_{e=0}^{I} z_e / C_{\tilde{Z}_e} , \qquad (22)$$

where $C_{\tilde{Z}_e}$ is composed of the left and right interval centroids $[\underline{c}_{z_e}, \bar{c}_{z_e}]$. The computation of the centroids of the zSlices can straightforwardly be obtained through any type-reduction method for IT2 fuzzy sets. In the present work, given the need for providing a rapid response and decoding from the brain signal, we use the Enhanced Iterative Algorithm with Stopping Condition (EIASC) [34], which allows faster processing than the standard Karnik-Mendel method [35] and has successfully been applied to real-time applications [36], [37]. $C_{\tilde{Z}}$ is a discrete set made of the centroids of all zLevels z_e. The zLevel z_0 is usually omitted because it represents samples that do not contribute to the GT2 fuzzy set. The centroid of the rule is defuzzified as a weighted average of the weighted outputs of the centroid of each zLevel as [33]:

$$v_j = \frac{\sum_{e=1}^{I} z_e \cdot C_{\tilde{Z}_j}(z_e)}{\sum_{e=1}^{I} z_e} . \qquad (23)$$

Likewise, we can find the membership grade of a ith sample via defuzzification:

$$\tilde{\psi}_j(x_i) = \frac{\sum_{e=1}^{I} z_e \cdot \tilde{Z}_j(x_i)}{\sum_{e=1}^{I} z_e} . \qquad (24)$$

The value of $\tilde{\psi}_j(x_i)$ is the membership of a sample to the GT2 fuzzy set linked to rule j. This parameter is an interval value $[\bar{\psi}_j(x_i), \underline{\psi}_j(x_i)]$. Recall that the zSlices are defined by their lower and upper bounds in Eq. (19). Then, $\tilde{\psi}_j(x_i)$ is type-reduced using the Nie-Tan method [38], resulting in $\psi_j(x_i)$. The pseudocode of the GT2FGG classifier is detailed in Table III.

TABLE III: GENERAL TYPE-2 FUZZY GATH-GEVA CLASSIFIER

	Input:	Vector x, trial labels l, fuzziness level $m \in M$, initial fuzzy partitions u, number of rules k.
	Output:	Rules v, Σ, u, p_j, f_j and $h_{j,k}$.
1	Repeat	
2		Estimate fuzzy centroid of each rule by Eq. (5).
3		Estimate fuzzy covariance of each rule by Eq. (6).
4		Compute degree of membership by the fuzzy partition function for all data samples by Eq. (4).
5		Estimate rule class associations by Eq. (10). Calculate a priori probability of the degrees of association by Eq. (3)
6		Compute distance from all samples to each rule by Eq. (11).
7		Estimate interval values of the primary function (20 and 21). Generate zSlices for each zLevel by Eq. (19).
8		Compute zSlices centroids by Eq. (23).
9		Compute membership grades by Eq. (24).
10		Update sum partition functions by Eq. (25) and (26).
11		If condition (7) is satisfied
12		Break loop.
13		end
14	End	

D. Self-Adaptation for GT2FGG

As the amount of training is limited in BMI, it is necessary to re-train our fuzzy model during the test periods. Similar to semi-supervised learning, the test data may serve to adapt our fuzzy model so as to cope with the changeable nature of the brain response across the course of the study. In addition to model adaptability based on parameter tuning, we propose a further self-adaptive methodology to adapt the whole structure of the model. For this purpose, several equations from Sections III.C and III.D must be updated to be able to adapt the model parameters as well as monitor the quality of the FRB in a sample-by-sample (i.e. trial-by-trial) basis. Thus, we define the sum of membership functions of all trials as:

$$\gamma_j^{N+1} = \gamma_j^N + \psi_j(x_{N+1}) , \qquad (25)$$

and the sum of the membership functions per class in each rule is represented by:

$$\gamma_{j,k}^{N+1} = \gamma_j^N + \psi_j(x_{N+1}) \, \delta(k^*, k), \qquad (26)$$

From parameters (25) and (26) the recursive update of equations (3) and (10) is straightforward:

$$p_j^{N+1} = \frac{1}{N+1}\gamma_j^{N+1}, \qquad h_{j,k} = \frac{\gamma_{j,k}^{N+1}}{\gamma_j^{N+1}} \quad . \tag{27}$$

Another model parameter that requires updating is the fuzzy mean, which can accommodate a new value as follows:

$$v_j^{N+1} = v_j^N + \frac{x_{N+1} - v_j^N}{\gamma_j^{N+1}}\psi_j(x_{N+1}) . \tag{28}$$

Likewise, the fuzzy covariance is updated with the new sample:

$$\Sigma_j^{N+1} = \frac{\gamma_j^N}{\gamma_j^{N+1}}\left(\Sigma_j^N + \frac{\psi_j(x_{N+1})}{\gamma_j^{N+1}}(x_{N+1}-v_j^N)(x_{N+1}-v_j^N)^T\right) \quad . \tag{29}$$

Note that this step does not require computing the centroids for all zLevels as in Eq. (23). The fuzzy covariance is updated using the resulting centroid membership of the new data sample, from which the inverse fuzzy covariance matrix can also be computed. Nowadays, robust solutions exist for computing matrix inversion and can be efficiently implemented using parallel programing [39]. As an alternative, the inverse covariance can also be updated recursively making use of the Sherman-Morrison-Woodbury inversion lemma for any non-singular square matrix (see [40-42]). Thus the update of the fuzzy precision matrix can be defined as:

$$(\Sigma_j^{N+1})^{-1} = \frac{\gamma_j^N}{\gamma_j^{N+1}}\left((\Sigma_j^N)^{-1} - \frac{(x_{N+1}-v_j^N)(x_{N+1}-v_j^N)^T}{\Sigma_j^N\left(\frac{\gamma_j^{N+1}}{\psi_j(x_{N+1})} + \frac{(x_{N+1}-v_j^N)(x_{N+1}-v_j^N)^T}{\Sigma_j^N}\right)\Sigma_j^N}\right) . \tag{30}$$

E. Online Model Self-Adaptive Development

First and foremost, adapting the FRB requires the definition of several conditions. These are based on different metrics that evaluate the state of the current model in relation to the new incoming data. To this end, we can use the online model parameters presented in the previous section, i.e. rule means and covariance. Three conditions are defined below in order to monitor the self-adaptive process of the learning algorithm:

1) Condition A: If this condition is satisfied then the new data sample is considered as a potential candidate to become the center of a new rule. Similarly to working with Gaussians, we can imagine that each rule has a hyper-ellipsoidal area of influence defined by the Mahalanobis similarity function as follows:

$$\zeta_{i,j} = \frac{1}{1+\sqrt{(x_i - v_j)^T\Sigma^{-1}(x_i - v_j)}} \quad . \tag{31}$$

The leading rule that exerts the highest influence over the new point is defined as:

$$j' = \arg\max_{j=1,\dots,R} \zeta_{i,j} \quad . \tag{32}$$

The minimum influence that a leader rule can have over a sample is formulated as follows:

$$\varphi_{j'} = \frac{\Gamma}{\Gamma + \lambda_{j'}^{-\partial}\chi_n^2(\alpha)(1+\lambda_{j'})^{\partial}} . \tag{33}$$

where $\lambda_{j'}$ is a counter of the number of times that rule j' comes out as leader rule. Eq. (33) uses the value of a chi-square (χ_n^2) distribution of n degrees of freedom to delimit a statistical tolerance region over the Mahalanobis distance. This is used as a scaling term in the divisor of a function of counter $\lambda_{j'}$, which permits rules with low backing to also preserve low minimum

influence. Γ is a scaling parameter, with recommended value equals to the dimensionality of the data. ∂ depicts a degree for the influence, which can simply be set to one. The inverse of the chi-square cumulative distribution function, with a desired confidence level α, can be estimated from any close approximation of the quantile function of the distribution, such as the Wilson-Hilferty method [43]. Finally, Condition A is defined as

$$A \rightarrow \zeta_{i,j} < \varphi_{j'} \tag{34}$$

2) Condition B: While Condition A was based on the local relationship between the new data point and an existing rule, Condition B can be developed to add a level of vigilance over the hypervolume coverage. In practice, we want the new rule to cover a different area of the hyperspace from our existing rule. Thus, we estimate that the dispersion of the leader rule will be affected by the new sample. Then, let us define this ratio of difference between these volumes as:

$$\tau_{j'} = \frac{\left\|\Sigma_{j'}^N\right\|}{\left\|\Sigma_{j'}^{N+1}\right\|} \tag{35}$$

where the symbol $\|\dots\|$ represents the volume of the fuzzy covariance function. A common way of determining the hyper-volume spanned by a covariance matrix is to use the determinant, albeit any other sophisticated methods for computing the volume of hyper-ellipsoids are also valid [44]. The values of τ have been shown to follow a beta distribution defined by $Beta(a,b)$, when $a = N - n - 1/2$ and $b = n/2$ [45]. A threshold for Eq. (35) can be set as the values of the inverse cumulative distribution specified by a and b with significance level α. Condition B is then expressed as:

$$B \rightarrow \tau_{j'} < Beta_\alpha(a,b) \tag{36}$$

3) Condition C: Last but not least, Condition C serves to correct the FRB and thus obtain a more compact and expressive set of rules. Then the overlap degree of every rule in the FRB is compared with the rest using the Bhattacharyya distance. The latter measures the closeness between two distributions, which can also be understood as the degree of overlap between the sample sets that both distributions represent [46]. Its equation for a multivariate normal distributions is:

$$\beta_{q'j'} = \frac{1}{8}\left(v_{j'} - v_{q'}\right)^T\Sigma_{j'+q'}^{-1}\left(v_{j'} - v_{q'}\right) + \frac{1}{2}\ln\left(\frac{\left\|\Sigma_{j'+q'}\right\|}{\sqrt{\left\|\Sigma_{j'}\right\|\left\|\Sigma_{q'}\right\|}}\right). \tag{37}$$

where $\Sigma_{q'+j'} = \left(\Sigma_{q'} + \Sigma_{j'}\right)/2$. Then, the rule with the largest overlap with i^* is defined by the following equation:

$$q' = \max(\beta_{q',j'}, \forall q \in \{1,\dots,R\}, q \neq j'). \tag{38}$$

If the value of $\beta_{q'j'}$ is greater than zero, we can consider it as a sign that both rules are collapsing. Therefore a merging operation of the rules represented by q' and j' can be performed, using the following equations:

$$\gamma_{q'j'} = \gamma_{q'} + \gamma_{j'} , \tag{39}$$

$$v_{q'j'} = \frac{\gamma_{q'}\,v_{q'} + \gamma_{j'}\,v_{j'}}{\gamma_{q'j'}} , \tag{40}$$

$$\Sigma_{q'j'} = \frac{1}{\gamma_{q'j'}^2}\begin{bmatrix}(\gamma_{q'}^3 + 2\gamma_{q'}^2\gamma_{j'} + \gamma_{q'}\gamma_{j'}^2)\Sigma_{q'} \\ + (\gamma_{j'}^3 + 2\gamma_{j'}^2\gamma_{q'} + \gamma_{j'}^2\gamma_{q'})\Sigma_{j'} \\ + (\gamma_{q'}\gamma_{j'}^2 + \gamma_{q'}^2\gamma_{j'})(v_{q'} - v_{j'})^T\end{bmatrix}. \tag{41}$$

Once the merge operation has been performed, an additional condition has been proposed in previous work to assess the homogeneity of the merged rules [47]. The differences in shape and orientation of the rules could result in the hypothetical case that the volume of the merged rule will cover an area that was not previously covered by rules q' and j'. To ensure that this issue does not occur, a convenient vigilance condition is to compare the new merged rule with the old ones as follows [47]:

$$\left\| \Sigma_{q'j'} \right\| \le n \left(\left\| \Sigma_{q'} \right\| + \left\| \Sigma_{j'} \right\| \right). \tag{42}$$

Finally, Condition C is given by:

$$C \rightarrow \beta_{q'j'} > 0 \ AND \ \left\| \Sigma_{q'j'} \right\| \le n \left(\left\| \Sigma_{q'} \right\| + \left\| \Sigma_{j'} \right\| \right) \tag{43}$$

When a new rule is generated, the new point x_{N+1} is considered the center of the rule $v_{R+1} = x_{N+1}$. If this is the first rule to be added to the rule base then it can be defined as $\Sigma_{R+1} = diag(range(X))$, otherwise it can be set as the average of the covariance of the already existing rules:
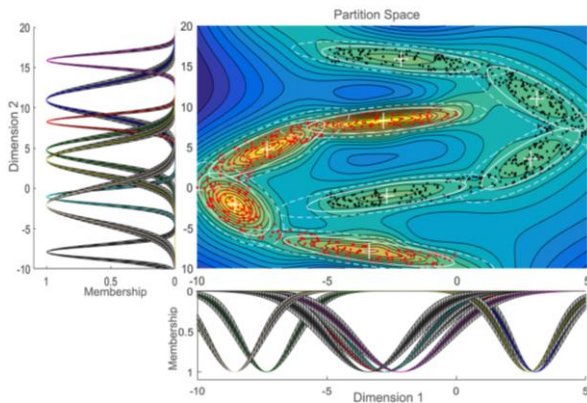

Fig. 2. Shows the partition space in the main colored chart along with the GT2 fuzzy membership functions on the left and bottom as viewed from the top. Shaded areas from grey to back display the grades across the secondary membership function. The partition function goes from blue (lowest) to yellow (highest) for all points in the space. The white markers (+) indicate the location of the centroids while the white ellipsoids represent their lower (dotted line) and upper (solid line) dispersions.

TABLE IV. RULE DESCRIPTIONS FOR THE EXAMPLE IN FIG. 2

	Centroid		Upper spread		Lower spread	
Rule	Dim. 1	Dim. 2	Dim.1	Dim.2	Dim.1	Dim. 2
1	3.0414	10.9608	1.2688	4.6907	0.645	2.3219
2	-7.3188	4.7448	1.4493	3.4232	0.7994	1.8642
3	-2.8355	8.2532	3.8999	0.837	2.4677	0.5135
4	-2.7138	-1.0634	2.9263	1.0347	1.7066	0.5962
5	-2.1728	15.8757	4.0258	0.8434	2.4186	0.4925
6	2.8185	3.5415	1.3385	4.4723	0.7328	2.3794
7	-3.3881	-7.9124	3.3178	0.9739	1.7443	0.5057
8	-8.5583	-2.0672	1.3364	5.9856	0.8249	3.4143

$$\Sigma_{R+1} = \frac{\sum_{j=1}^{R} \Sigma_j}{R} \tag{44}$$

Fig. 2 and Table IV give an example of application of GT2FGG for a partition space with convex "banana shape" patterns.

F. Online Feature Ranking and Rule Re-Scaling

An important part of the BMI literature deals with the issue of determining which variables from the input data provide better discriminability for the target mental task. This is important to identify the channel or channel pair selection that best captures the underlying cortical activity patterns of the task. This is more challenging in the case of completely novice users given that the configuration may change across the course of the experiment whilst users stabilize their brain activity patterns based on the neurofeedback. Therefore, we provide a method for addressing this process incrementally. A practical indicator of class separability is given by the F-ratio criterion function, which computation is formulated as:

$$F(x) = \frac{tr(Q_b)}{tr(Q_w)}, \tag{45}$$

where Q_b and Q_w are the between- and within- scatter matrices, respectively. Intuitively, the idea is that the class separability capacity of a feature can be described by how much it maximizes $tr(Q_b)$ whilst minimizing $tr(Q_w)$. Additionally, the trace of a matrix is an easy calculation. From the computation of GT2FGG, we indeed have all the necessary ingredients to estimate Q_b and Q_w as an additional online processing step. The standard definition of Q_w is the sum of the covariances from all subgroups, i.e. $Q_w = \Sigma_1 + \Sigma_2$ [48]. In our model, as the samples from one class can be subdivided into several rules, we then modify the standard definition into the following equation:

$$Q_w^h = \sum_{k=1}^{K} \sum_{j=1}^{R} \frac{h_{j,k} p_j \Sigma_j^{h'}}{\tilde{h}_k}, \tag{46}$$

where $\tilde{h}_k = \Sigma_{j=1}^{R} h_{j,k}$, η is an index indicating a feature, and η' is the set of all features distinct to η. In Eq. (46), the covariance matrix of each rule is weighted by the normalized degree of class association and the expected a priori degree of association of a rule. As for the covariance matrix, only the rows and columns of the dimensions $\eta \ne \eta'$ are considered. The between scatter matrix is typically defined as $Q_w = p_1(v_1 - v_0)(v_1 - v_0)^T + p_2(v_2 - v_0)(v_2 - v_0)^T$ [48], where v_0 is the global mean. In the present work, we define the global mean as the mean from all rule centers:

$$v_0 = \frac{1}{R} \sum_{j=1}^{R} v_j, \tag{47}$$

and the class mean as:

$$v_k = \sum_{j=1}^{R} \frac{h_{j,k} v_j}{\tilde{h}_k}. \tag{48}$$

Thus, we can define our between scatter as follows:

$$Q_b^\eta = \sum_{k=1}^{K} \frac{\tilde{h}_k S_k^{\eta'}}{\tilde{\tilde{h}}}, \tag{49}$$

where $S_k = (v_k - v_0)(v_k - v_0)^T$ and $\tilde{\tilde{h}} = \Sigma_{k=1}^{K} \tilde{h}_k$ for $\eta' \subset (\{1,...,n\} \ne \eta)$ rows and columns in the covariance. From Eq. (46) and (47), we have the necessary parameters to compute (45). Note that Eq. (46) and (49) reuse the parameters obtained during the learning process of GT2FGG, so that it is not necessary to loop over all the training samples as with the original computation of the between- within- scatter matrices. Alternatively, for computational speed improvements, the update of the scatter matrices can be performed by computing the parameter differences with:

$$\Delta h_{j,k^*} = h_{j,k^*}^{N+1} - h_{j,k^*}^{N}, \quad \tilde{h}_{k^*}^{N+1} = \tilde{h}_k^N + \Delta h_{j,k^*}, \quad \tilde{\tilde{h}}^{N+1} = \tilde{\tilde{h}} + \Delta h_{j,k^*}, \tag{50}$$

$$\Delta \Sigma_j = \Sigma_j^{N+1} - \Sigma_j^N, \quad \Delta S_{k^*} = S_{k^*}^{N+1} - S_k^N. \tag{51}$$

Note that the computation of the differences is only needed for the leader rule and the predicted class. Thus, an incremental update of the scatter matrices can be performed as:
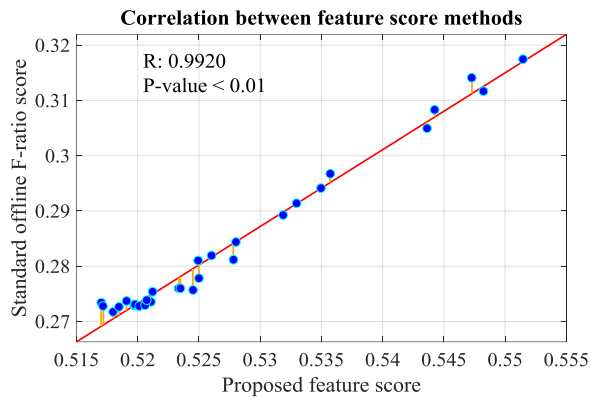
Fig. 3. Shows the correlation between the standard F-ratio and the proposed method for a synthetic dataset. The synthetic test dataset with 200 samples was generated using two multivariate Gaussian distributions with different means as well as a diagonal covariance matrix with thirty dimensions and variances equal to 0.5. Gaussian noise was added to each dimension in scale from 1 to 30 signal-to-noise ratio per sample. The resulting scores are significantly and positively correlated [R=0.9932, p < 0.01 (Pearson)].

$$Q_w^{\eta, N+1} = Q_w^\eta + \frac{1}{\tilde{h}_{k^*}^{N+1}} \cdot h_{j',k^*}^{N+1} \cdot P_{j'}^{N+1} \cdot \Delta \Sigma_{j'}^{\eta'} \qquad (52)$$

$$Q_b^{\eta, N+1} = Q_b^\eta + \frac{1}{\tilde{\tilde{h}}^{N+1}} \cdot \tilde{h}_{k^*}^{N+1} \cdot \Delta S_{k^*}^{\eta'} \qquad (53)$$

Fig. 3 shows a comparison between the proposed method and a standard computation of the F-ratio that runs over the whole set of training samples. Although the resulting scores differ in magnitude, they exhibit a very high significant positive correlation. Ergo, our method supplies, for all practical purposes, a tantamount feature score.

As formulated in (45), the value of the ratio should get higher for meaningful features. However, as specified in the inequality in (48) and (47), we are leaving out the queried feature to be able to assess the impact of the removal of this feature on the F-ratio. Thus, the smaller the resulting F-ratio, the higher the relevance of the feature. For this reason, we take the inverse of the F-ratio to obtain the opposite trend and the resulting weight is viewed as a proportion of the feature with the maximum score. Therefore, the weight for feature η is defined as:

$$\theta^\eta = \frac{tr(Q_w^\eta) / tr(Q_b^\eta)}{\max\limits_{\eta=1,\dots,n}\left\{ tr(Q_w^\eta) / tr(Q_b^\eta) \right\}}, \qquad (54)$$

Once a feature weight is obtained, it can be used to re-scale the covariance matrix so that the features with low weights will have a smaller effect on the computation of the Mahalanobis similarity function in (31), as proposed in [49]. A common approach is to remove the low weighting features from the system. Alternatively, a practical method presented in [49] is to use these weights for penalizing the less important features during the model construction, i.e. the weights can serve to perform a shrinkage of the covariance matrix. This scaling is performed as follows:

$$\Sigma_j^{scaled} = V \Theta D \Theta V^T \qquad (55)$$

where V comes from the singular value decomposition $\Sigma = VDV^T$ and Θ is the resulting mapping of the feature weights into the subspace spanned by the covariance of the rule as $\Theta = diag(V^T diag(\theta) V)$. This scaling can be performed both with the covariance and its inverse. This scaled covariance with

feature restrictions can be used in following operations involving these matrices, thus lowering the impact of redundant features without having to remove them.

The pseudocode of the self-adaptive GT2FGG classifier is detailed in Table V along with a flowchart diagram in Fig. 4.
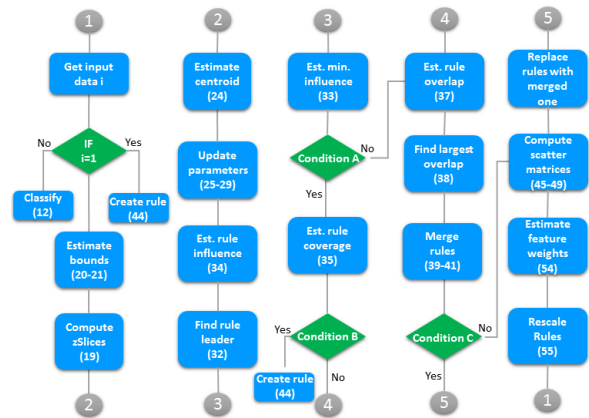

Fig 4. A flowchart diagram illustrating the overall processing steps of the self-adaptive GT2FGG classifier.

TABLE V: ONLINE SELF-ADAPTIVE GENERAL TYPE-2 FUZZY GATH-GEVA CLASSIFIER

	Input:	Test input x, fuzziness level $m \in M$.
	Output:	Rules v, Σ, u, p_j, f_j and $h_{j,k}$.
1	*Repeat*	
2		Classify test sample x by Eq. (12).
3		Estimate upper and lower memberships for each zLevel by Eq. (20) and (21).
4		Add the zSlices for each zLevel by Eq. (19).
5		Compute membership grade by Eq. (24).
6		Incrementally update model parameters by Eq. (25), (26), (27), (28), and (29).
7		Compute hyper-ellipsoidal area of influence by Eq. (31).
8		Find leader rule by Eq. (32).
9		Compute minimum tolerated influence by Eq. (33).
10		*If* Condition A (34) is satisfied *Then*
11		Compute ratio of coverage by Eq. (35).
12		*If* Condition B (36) is satisfied *Then*
13		Create new rule.
14		*End*
15		*end*
16		Compute measure of overlap by Eq. (37).
17		Find rule with largest overlap by Eq. (38).
18		Merge rules by Eq. (39), (40) and (41).
19		*If* Condition C (43) is satisfied *Then*
20		Remove rules q' and j', and keep their merged one
21		*end*
23		*If* feature weighting is active *Then*
24		Compute scatter matrices by Eq. (45–49)
25		Estimate feature weights by Eq. (54)
26		Rescale feature weights by Eq. (55)
27		*end*
28	*End*	

IV. EXPERIMENTS AND SETTINGS

A. EEG Settings and Filters

The EEG signal was recorded using an EEG system with active electrodes as well as a BrainAMP DC amplifier (Brain Products GmbH). Only six electrodes were used as per the positions shown in Fig. 5. AFz and FCz were used a ground and reference electrodes. The EEG signal frequency was 500 Hz per

channel. The users were asked to remain still and limit eye movements to avoid electrooculography (EOG) artifacts. Additionally, the FORCe method [50] was employed for online artefact removal on the training dataset. Trials with amplitude exceeding ±300 uV were discarded. A posterior visual examination of the recordings was performed to confirm the absence of artifacts. A band-pass (4th order Butterworth filter) was applied between 0.5 and 50Hz.

B. BMI Experiments

Fifteen users (12 men and three women; mean age 27.33 ± 4.8 years old; two left handed), untrained in BMI, were recruited to participate in the experiment. A programmable humanoid robot (Nao, Aldebaran Robotics) was used as the brain controlled device. Local research ethics committee approval was obtained (project number: 05/Q0403/142). Two different BMI learning approaches were implemented, namely a basic screen-based cue task (referred to as "basic task") followed by an online robot navigation with a Nao Robot (referred to as "robot task"):

1) Basic task (Fig. 6.1): A visual stimulus is displayed, highlighting a pointer symbol either left or right, upon which the user is given two seconds to perform a sustained MI task, i.e., for either right or left motion. Then, a relaxation period is allowed before the next iteration. Thus, each trial consists of

moderate confidence level, those around 0.8 a high confidence level, and those around 0.9 a very high agreement. For distorted confidence scores, the Platt's scaling is used [51]. A system update is performed when the confidence level prediction reaches values higher than 0.8. Then, a short sound is produced and the robot proceeds with the corresponding action. The BMI commands and signals are annotated and recorded for posterior analysis.

C. EEG Pre-Processing

A large majority of previous BMI approaches have been developed for features generated via CSP filters [52], which decompose the original EEG signal into a set of additive subcomponents, also called "bands". In CSP, the feature generation requires knowing the covariance of the respective classes in advance in order to develop the subsequent spatial transformations of the features that provide maximum separability between the classes.

A limitation of this approach is that, in order to apply CSP, a set of class labels must be provided a priori and, when the users are untrained, the class labels from the initial session can be unreliable. Moreover, in cases where the goal is to move from a basic BMI paradigm with visual cues on the screen to more demanding scenarios that involve controlling a robot in real-time, the users' brain responses may exhibit non-stationary signal as a result of this context change. Then, a supervised
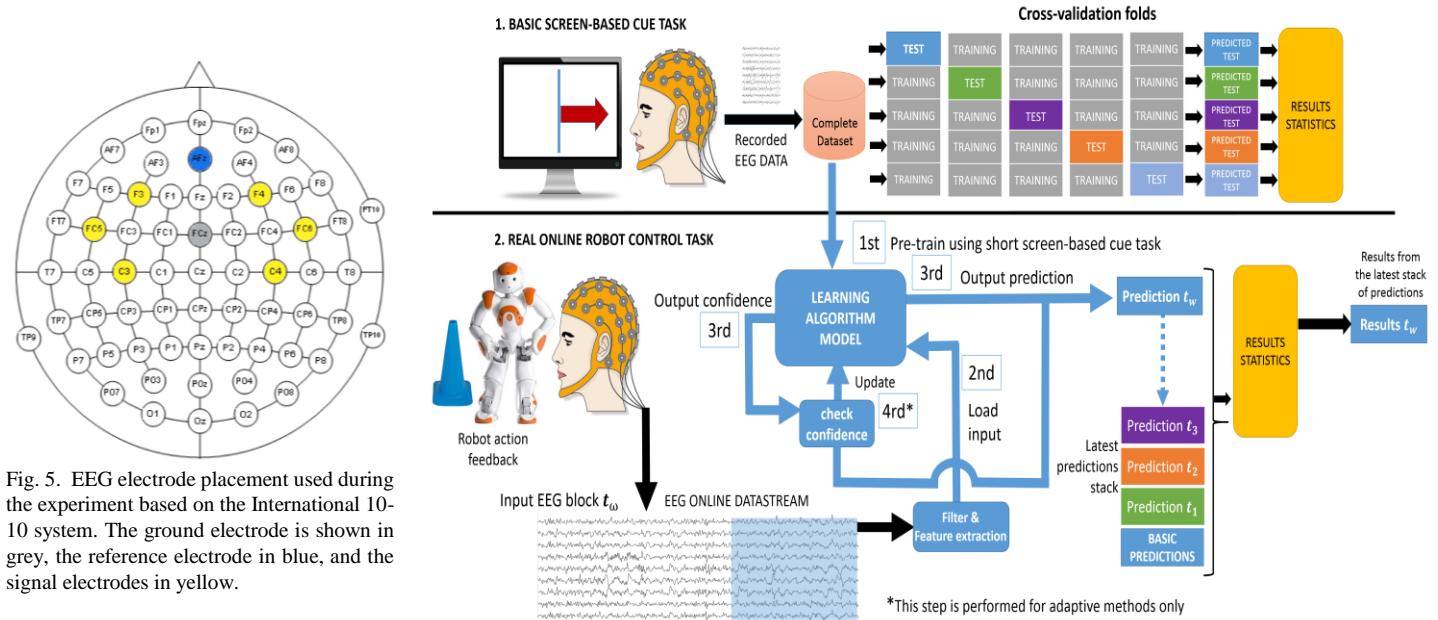

Fig. 5. EEG electrode placement used during the experiment based on the International 10-10 system. The ground electrode is shown in grey, the reference electrode in blue, and the signal electrodes in yellow.

Fig. 6. Processing schema and evaluation for the basic and robot tasks.

one guided MI task performed by the user. For the first task, the visual cues are displayed in a random order with the condition that 10 trials of each MI task (i.e. left or right) are presented without any feedback.

2) Robot task (Fig. 6.2): In this task, the user is initiated to a humanoid robot control, which continuously and slowly moves forward in a 25m^2 empty space. The user is instructed to turn the robot left or right using MI towards the opposite side and a blue cone is intentionally placed on the robot trajectory. The user continues performing MI until the obstacle is avoided or removed. As regards the predictions, confidence values around 0.6 indicate a poor confidence level, those around 0.7 a

feature extraction method may add another subjacent source of decalibration, in addition to the one experimented by the classifier.

Features based on spatial filters such as surface Laplacian, are also frequently used in the BMI literature [53]. This method aims to estimate a radial current at the scalp, using the recorded EEG signal. Although this does not require class labels, for accurate estimations, it may involve operations of high computational complexity such as spherical spline interpolations. In the present work, we obtained the highest accuracy using phase synchrony features, which are class-free

TABLE VI. PERFORMANCE COMPARISON

Acc.: Accuracy; #R: number of rules or length of basic structure; T'': seconds taken for retraining or sequential learning from one sample instance with Intel proc. i7 quad-core @3.4 GHz; s.v: support vectors; McIT2FIS: metacognitive interval type-2 fuzzy; oLDA: online linear discriminant analysis; AdIM: adaptive information matrix; DBN: deep belief nets; SCGNF: scaled conjugate gradient neuro fuzzy; ARTMAP: adaptive resonance theory mapping; T1FGG: type-1 fuzzy Gath-Geva classifier; RF: random forests; RBFSVM: radial basis function support vector machine; LinSVM: linear support vector machine; NN: neural-networks; T1FSIMclass: type-1 fuzzy simpl_eclass; SA-GT2FGG: Self-Adaptive General Type-2 Fuzzy Gath-Geva.

| Learning Method | Used in BMI | Type | Basic task | | | | Robot Task | | | | |
			Acc.%	TP rate	TN rate	#R	Acc. %	TP rate	TN rate	#R	T''
*SA-GT2FGG (Proposed)	Yes	GT2 Fuzzy	85.11 ± 3.30	0.86 ± 0.10	0.84 ± 0.07	5.1 ± 0.8 rules	85.78 ± 2.05	0.90 ± 0.05	0.81 ± 0.05	6.7 ± 0.7 rules	0.0026 ± 0.0002
*McIT2FIS [22]	Yes	IT2 Fuzzy	62.05 ± 10.42	0.63 ± 0.144	0.61 ± 0.117	6.3 ± 1.58 rules	61.49 ± 9.85	0.78 ± 0.148	0.45 ± 0.201	8.9 ± 0.8 rules	0.0025 ± 0.0002
*oLDA [3] [18, 55]	Yes	Linear discrit.	52.81 ± 12.16	0.77 ± 0.139	0.30 ± 0.181	2 cov.	58.38 ± 9.96	0.65 ± 0.133	0.51 ± 0.149	1 cov.	0.0033 ± 0.0001
*AdIM [3]	Yes	Quadratic discrit.	62.00 ± 7.95	0.70 ± 0.118	0.54 ± 0.122	2 cov.	69.63 ± 7.25	0.78 ± 0.079	0.61 ± 0.103	2 cov.	0.0014 ± 0.0005
*SA-GT2FGG (no feat. scaling)	No	GT2 Fuzzy	80.20 ± 3.67	0.81 ± 0.072	0.80 ± 0.045	6.08 ± 1.09 rules	82.52 ± 4.23	0.79 ± 0.051	0.87 ± 0.059	6.9 ± 1.62 rules	0.0014 ± 0.0002
*SA-GT2FGG (with T1 Fuzzy)	No	T1 Fuzzy	69.80 ± 8.31	0.70 ± 0.097	0.70 ± 0.087	9.26 ± 0.96 rules	73.45 ± 4.47	0.72 ± 0.040	0.76 ± 0.063	10.4 ± 1.76 rules	0.0022 ± 0.0002
*T1FSIMclass [56, 57]	No	T1 Fuzzy	69.85 ± 9.36	0.70 ± 0.088	0.71 ± 0.116	8.93 ± 2.52 rules	68.78 ± 5.81	0.68 ± 0.063	0.70 ± 0.068	14.2 ± 2.04 rules	0.0019 ± 0.0001
DBN [58]	No	NN	55.66 ± 5.64	0.60 ± 0.219	0.45 ± 0.467	100 neurons	50.44 ± 6.41	0.40 ± 0.161	0.66 ± 0.439	100 neurons	0.6013 ± 0.01
SCGNF [59] [60]	No	Fuzzy NN	71.08 ± 7.00	0.68 ± 0.17	0.68 ± 0.15	10 neurons	65.46 ± 12.19	0.69 ± 0.109	0.77 ± 0.153	10 neurons	0.2628 ± 0.04
Fuzzy ARTMAP [61]	No	Fuzzy ART	71.75 ± 7.75	0.68 ± 0.173	0.69 ± 0.152	10 rules	67.46 ± 13.83	0.67 ± 0.15	0.74 ± 0.17	10 rules	0.2897 ± 0.05
T1FGG [30]	No	T1 Fuzzy	60.19 ± 7.85	0.62 ± 0.142	0.62 ± 0.137	10 rules	60.74 ± 7.58	0.60 ± 0.121	0.59 ± 0.146	10 rules	0.019 ± 0.06
RF [62]	Yes	Ensemble Trees	74.68 ± 7.69	0.70 ± 0.149	0.79 ± 0.093	10 trees	67.33 ± 9.10	0.64 ± 0.125	0.71 ± 0.127	10 trees	0.1195 ± 0.014
RBFSVM [63]	Yes	Non-linear SVM	66.34 ± 10.64	0.56 ± 0.212	0.66 ± 0.191	9 ± 1 s.v.	61.15 ± 12.84	0.66 ± 0.162	0.67 ± 0.186	9 ± 1 s.v.	0.0089 ± 0.001
LinSVM [64]	Yes	Linear SVM	63.57 ± 11.90	0.58 ± 0.213	0.64 ± 0.201	30.18 ± 2.14	62.30 ± 12.44	0.67 ± 0.188	0.63 ± 0.202	30.18 ± 2.14	0.0086 ± 0.0009

and their computation is nothing more than an analysis of the phase differences between the signals [54]. The phase synchrony features are computed from three pairs of channels displayed in Fig. 5, namely F3-F4, FC5-FC6, and C3-C4. The location of these pairs provides us with enough EEG coverage of the motor cortex. Then, the following steps are performed:

1) Step 1. For each raw EEG signal from a specific channel ι denoted by $\varpi^\iota(t)$, every time the buffer gets filled for a period of length Ω, we compute the Hilbert transform of every signal at time t as $\tilde{\varpi}^1(t) = H(\varpi^1(t))$ [65].

2) Step 2. Following step 1, we derive its Hilbert phase as:

$$\phi^1(t) = \arctan \frac{\tilde{\varpi}^1(t)}{\varpi^1(t)} ; \qquad (56)$$

3) Step 3. Finally, a value of phase synchronization can be estimated by the mean phase difference between two signals [66, 67]:

$$x^{1,2} = \left| \frac{1}{\Omega} \sum_{\omega=1}^{\Omega} e^{\kappa[\phi^1(t_\omega) - \phi^2(t_\omega)]} \right| , \qquad (57)$$

where $\kappa = \sqrt{-1}$ and x is one dimension of the input used for the learning model.

V. PERFORMANCE ANALYSIS

The result section consists of three analyses: A) performance results of the proposed method along with 14 other algorithms on the recorded data from both tasks; B) a statistical analysis of results comparison between the methods; C) performance results of the proposed method using different features.

A. Performance Results

For the basic task, the classifiers were allowed to be trained offline using a 5-fold cross-validation process. The predictions from each fold were copied in their corresponding trial order into a global prediction vector of size equal to the total number of trials in the original session of the corresponding user. This organization of the predictions enables to obtain a total evaluation metric per user, rather than averaged results from all folds. As the robot task is performed online, predicted values are recorded for each sequential trial and these are then compared against the ground truth annotations of the suggested actions at the end of the experiment.

Several machine learning algorithms were tested to evaluate the adequacy of SA-GT2FGG for performing adaptive learning. We compared their accuracy, both during the basic and robot tasks.

TABLE VII. COMPARISON OF ACCURACY BETWEEN SA-GT2FGG AND ALTERNATIVE METHODS - BASIC TASK

Method A	Method B	Test Estimate	P-Value $\alpha=0.05$
SA-GT2FGG	oLDA	0.03	**p<0.01**
SA-GT2FGG	AdIM	0.15	**p<0.01**
SA-GT2FGG	McIT2FIS	-0.01	**p<0.01**
SA-GT2FGG	SA-GT2FGG-NS	0.03	**p<0.01**
SA-GT2FGG	SA-GT2FGG-T1FS	0.03	**p<0.01**
SA-GT2FGG	T1FSIMclass	0.07	**p<0.01**
SA-GT2FGG	DBN	0.13	**p<0.01**
SA-GT2FGG	SCGNF	0.09	**p<0.01**
SA-GT2FGG	ARTMAP	-0.01	**0.01**
SA-GT2FGG	T1FGG	0.02	**p<0.01**
SA-GT2FGG	RF	0.01	**p<0.01**
SA-GT2FGG	RBFSVM	0.2	**p<0.01**
SA-GT2FGG	LinSVM	-0.03	**p<0.01**

TABLE VIII. COMPARISON OF ACCURACY BETWEEN SA-GT2FGG AND ALTERNATIVE METHODS - ROBOT TASK

Method A	Method B	Test. Estimate	P-Value $\alpha=0.05$
SA-GT2FGG	oLDA	0.02	**p<0.01**
SA-GT2FGG	AdIM	0.16	**p<0.01**
SA-GT2FGG	McIT2FIS	0.07	**p<0.01**
SA-GT2FGG	SA-GT2FGG-NS	0.09	**p<0.01**
SA-GT2FGG	SA-GT2FGG-T1FS	0.08	**p<0.01**
SA-GT2FGG	T1FSIMclass	0.13	**p<0.01**
SA-GT2FGG	DBN	0.16	**p<0.01**
SA-GT2FGG	SCGNF	0.08	**p<0.01**
SA-GT2FGG	ARTMAP	-0.02	**p<0.01**
SA-GT2FGG	T1FGG	0.07	**p<0.01**
SA-GT2FGG	RF	0.1	**p<0.01**
SA-GT2FGG	RBFSVM	0.29	**p<0.01**
SA-GT2FGG	LinSVM	0.07	**p<0.01**

All algorithms were allowed to use the data from the basic task as training data. In addition, adaptive/self-adaptive methods were allowed to update their models. Evaluating offline and online methods enables to identify whether offline classifiers are able to generalize well to the online robot control paradigm without the need for online adaptation. We added two alternatives of the proposed algorithm, one with the feature scaling deactivated and one that uses type-1 fuzzy sets rather than type-2. Some algorithms such as LDA and SVMs are quite popular in BMI, while random forests and ARTMAP are offline methods able to generalize well despite learning with small training sets.

All methods were trained using phase synchrony features. As regards those methods that required the definition of parameters a priori, these were set-up to their optimal configurations as suggested in their original articles.

In Table VI, a set of algorithms are compared using inter-subject statistical metrics of performance:

$$Acc = \frac{tp + tn}{tp + fp + fn + tn}; \quad TPR = \frac{tp}{tp + fn}; \quad TNR = \frac{tn}{tp + fn}; (58)$$

where tp and tn are the numbers of true positives and true negatives, fp is the number of false positives, and fn is the number of false negatives.

B. Statistical Comparison of Results

A statistical analysis was performed to evaluate group differences between methods from all subject results in each task. Post-hoc analysis was performed after significant ANOVA (Welch's F test). The Shapiro-Wilk test results with alpha set to 0.05 can be found in the supplementary material (SP).

As regards the *basic task*, the post-hoc test results after significant ANOVA (F-value: 28.327, p < 0.01) are shown in Table VII (see complete table in SP). Thus, the results obtained with the proposed SA-GT2FGG method are significantly better than the ones obtained with the alternative methods, suggesting that, even in the situation of a basic screen-based cue BMI approach, SA-GT2FGG provides competitive performance with respect to the other methods. In addition, the results show that oLDA, AdIM, and McIT2FIS achieve a performance that

is significantly above poorly performing methods in this scenario such as DBN. However, their accuracy varies in significant difference with respect to other methods such as SCGNF, T1FGG, ARTMAP, RF, RBFSVM and LinSVM.

For the *robot task*, the detailed values of the post-hoc multi-comparison tests are provided in Table VIII (see complete table in SP) after significant ANOVA test (F-value: 61.8163, p < 0.01). Thus, we can observe that there is statistical difference in performance between the proposed SA-GT2FGG method and the other methods. Overall, there is some improvement using adaptive methods applied to BMI (oLDA, AdIM, McIT2FIS) during the robot task, although this is not sufficient to become statistically notable with respect to the other methods.

Finally, a Receiver Operating Characteristic (ROC) curve analysis is presented in Fig. 7 and Table IX for all the subjects studied, showing the performance of each method as its prediction confidence score (used feedback and adaptation) varies. We can observe that, while adaptive and linear methods such as oLDA and AdIM perform well (above the mid reference line) for several subjects, SA-GT2FGG still exhibits greater separability. Assigning maximum penalization for false positives, the optimal confidence threshold for SA-GT2FGG lies on 0.8254 ± 0.0249, which is right within the high confidence limit.

C. Signal Features and Settings Comparison

A performance analysis of SA-GT2FGG using signals from three different channel configurations and four feature extraction methods is implemented in this section. In order to compute CSP and band power features, a band-pass 8-30Hz filter is applied on the EEG signal a priori. Filter Band CSP takes as input a decomposition of the signal on eight frequency bands 4-8Hz, 8-12Hz…, subsequently up to 36-40Hz. Table IX displays the inter-subject results. Therefore, the analysis shows that SA-GT2FGG in conjunction with phase synchrony provides the best results, which is promising considering a very fast unsupervised feature extraction method such as phase synchrony.

VI. DISCUSSION

In this work, a novel BMI learning methodology was proposed to address the challenges raised by real online BMI systems with limited training data and changing context. In particular, two essential technical contributions can be highlighted: 1) implementation of GT2 FISs into a BMI experiment with real robot control using unsupervised signal features; 2) a novel GT2 fuzzy logic classifier that is able to self-develop by adapting its parameters and structure (number of rules) to accommodate to the very likely signal nonlinearities. GT2 fuzzy clustering stands on the foundations of GG likelihood dissimilarity and a rule-to-class association measure. The fuzzy inference of the proposed model was performed by applying the center-of-gravity principle of the rule's firing level and rule-class association, i.e. no rule is specifically bounded to just one class. Each rule was composed of GT2 fuzzy sets, which zSlices were formed by the upper and lower fuzzy memberships with respect to a fuzzy soft partitioning with the fuzziness degree of a determined zLevel (19-21). As a difference to methods based on IT2 fuzzy sets, employing GT2 fuzzy sets removes the need for keeping two mirrored processing stacks for the upper and lower memberships (e.g. keeping track of upper $\overline{\Sigma}$ and lower $\underline{\Sigma}$ covariance matrices). Once all the general fuzzy sets are type-reduced and defuzzified, the resulting centroids are regarded as rule centers. As for model updating and self-adaptive, an online monitoring method was presented based on a set of conditions that surveil the minimum influence exerted by the current model over a new sample, as well as the maximum coverage, and rule redundancy. The update of the fuzzy model parameters was then performed with incremental operations.

The achieved performance was high for both the basic screen-based cue and online robot tasks using phase synchrony features. The basic task used for training only consisted of 10 trials per class, meaning that performance in the robot task was fully dependent on adaptation. By the end of the experiment, all novice participants were able to perform the robot task accurately. Comparison results against T1 fuzzy and IT2 fuzzy methods indicate that scaling towards GT2 fuzzy models is an

structure may penalize the evolving learning flexibility required for the pursued objectives of the proposed research.
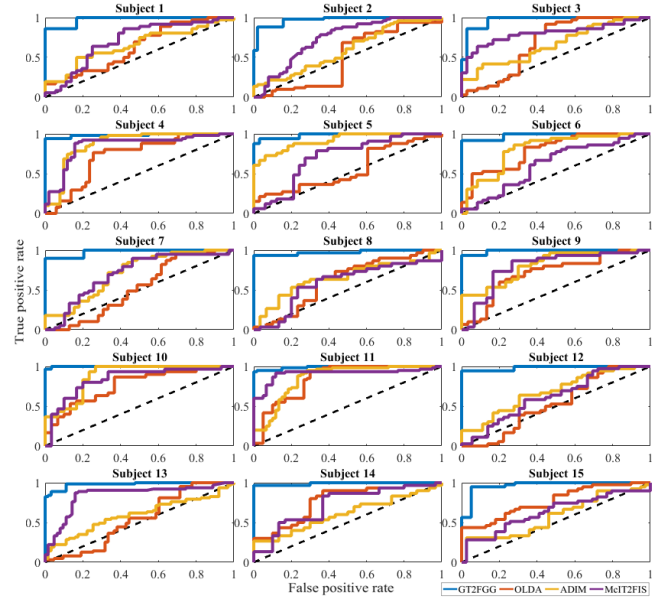

Fig. 7. ROC and Area Under the Curve (AUC) during the robot task for each subject.

TABLE IX. AREA UNDER THE CURVE STATISTICS (ROBOT TASK)

	SA-GT2FGG	oLDA	AdIM	McIT2FIS
Subject 1	0.98	0.63	0.66	0.71
Subject 2	0.97	0.5	0.6	0.74
Subject 3	0.97	0.68	0.67	0.8
Subject 4	0.98	0.73	0.87	0.85
Subject 5	0.98	0.56	0.92	0.71
Subject 6	0.98	0.8	0.81	0.62
Subject 7	0.98	0.53	0.72	0.71
Subject 8	0.97	0.62	0.65	0.59
Subject 9	0.99	0.71	0.81	0.79
Subject 10	0.99	0.77	0.88	0.84
Subject 11	0.99	0.84	0.85	0.91
Subject 12	0.98	0.53	0.68	0.62
Subject 13	0.98	0.54	0.59	0.84
Subject 14	0.99	0.77	0.61	0.73
Subject 15	0.97	0.77	0.58	0.63

VII. CONCLUSION AND FUTURE WORK

TABLE X. ACCURACY COMPARISON ACROSS DIFFERENT FEATURES AND SETTINGS WITH SA-GT2FGG

| Channels | Basic Task | | | | Robot Task | | | |
	PS	CPS (4 bands)	FBCSP (4 bands)	BP	PS	CSP (4 bands)	FBCSP (4 bands)	BP
4 channels (F3, F4, C3, C4)	75.64 ± 2.61	51.30 ± 2.61	74.53 ± 2.54	50.61 ± 1.44	74.64 ± 1.69	49.66 ± 1.91	71.24 ± 1.17	47.43 ± 3.47
6 channels (F3, F4, FC5, FC6, C3, C4)	85.11 ± 3.30	64.50 ± 3.41	75.54 ± 1.88	50.37 ± 2.08	85.78 ± 2.05	61.00 ± 2.29	72.20 ± 1.38	48.48 ± 1.97
8 channels (F3, F4, FC5, FC6, C3, C4, CP4, CP6)	72.02 ± 2.60	42.09 ± 5.87	73.23 ± 2.11	50.69 ± 1.55	72.14 ± 1.32	41.50 ± 6.53	70.62 ± 1.44	50.67 ± 1.16

PS: phase synchrony; CPS: common spatial pattern; FBCSP: Filter band common spatial pattern; BP: band power.

interesting option to improve performance. As to offline methods such as SVM and random forests, these seem to have failed to generalize to the robot task without adaptation capabilities. As regards popular adaptive methods in BMI such as oLDA and AdIM, the performance improved when adaptation was enabled, although they did not reach levels similar to the one obtained with SA-GT2FGG. While they remain relevant approaches due to their simplicity, their fixed

In conclusion, this work represents a step further in GT2 fuzzy systems by implementing a learning model based on this extended fuzzy logic framework for adaptive BMI systems. As regards the experiment, in future research we will work on reducing the amount of supervised training even further. In the current robot task, the user is constrained to perform MI at specific controlled timings and within a set of determined commands. Therefore, we will aim at extending the experimentation towards asynchronous and free control.

Likewise, we look forward to working with different control scenarios including rehabilitation devices and prostheses. As regards fuzzy methods, an area for future research would be to further explore the suitability and applicability of the type-reduction method for zSlice type-2 fuzzy such as centroid-flow algorithms. As for the FRB identification system, although we are satisfied with the trade-off between performance and complexity provided by our method for real-time operation, the self-adaptive structure could be extended to other applications and experiments by adding a rule-splitting condition in order to improve the specificity of the rules to the pattern distribution, at the cost of adding some extra computations.

REFERENCES

[1] C.-T. Lin, I.-F. Chung, L.-W. Ko, Y.-C. Chen, S.-F. Liang, and J.-R. Duann, "EEG-based assessment of driver cognitive responses in a dynamic virtual-reality driving environment," *IEEE T. Bio-med Eng.,* vol. 54, pp. 1349-1352, 2007.

[2] C.-T. Lin, Y.-C. Chen, T.-Y. Huang, T.-T. Chiu, L.-W. Ko, S.-F. Liang*, et al.*, "Development of wireless brain computer interface with embedded multitask scheduling and its application on real-time driver's drowsiess detection and warning," *IEEE T. Bio-med Eng.,* vol. 55, pp. 1582-1591, 2008.

[3] C. Vidaurre, A. Schlögl, R. Cabeza, R. Scherer, and G. Pfurtscheller, "Study of on-line adaptive discriminant analysis for EEG-based brain computer interfaces," *IEEE T. Bio-med Eng.,* vol. 54, pp. 550-556, 2007.

[4] B. Blankertz, C. Sanelli, S. Halder, E. Hammer, A. Kübler, K.-R. Müller*, et al.*, "Predicting BCI performance to study BCI illiteracy," *BMC Neurosci.,* vol. 10, p. P84, 2009.

[5] C. Guger, S. Daban, E. Sellers, C. Holzner, G. Krausz, R. Carabalona*, et al.*, "How many people are able to control a P300-based brain–computer interface (BCI)?," *Neurosci. Lett.,* vol. 462, pp. 94-98, 2009.

[6] J. d. R. Millán, "On the need for on-line learning in brain-computer interfaces," in *Proceedings of the IEEE International Joint Conference on Neural Networks.*, 2004, pp. 2877-2882.

[7] A. Nijholt and D. F. Tan, "Brain-computer interfacing for intelligent systems," *IEEE Intell. Syst.,* vol. 23, pp. 72-79, 2008.

[8] G. Pfurtscheller and C. Neuper, "Motor imagery and direct brain-computer communication," *Proc. IEEE,* vol. 89, pp. 1123-1134, 2001.

[9] A. D. Torshizi and M. H. F. Zarandi, "A new cluster validity measure based on general type-2 fuzzy sets: application in gene expression data clustering," *Knowl-Based Syst.,* vol. 64, pp. 81-93, 2014.

[10] O. Linda and M. Manic, "General type-2 fuzzy c-means algorithm for uncertain fuzzy clustering," *IEEE T Fuzzy Syst,* vol. 20, pp. 883-897, 2012.

[11] J. d. R. Millán and J. Mouriño, "Asynchronous BCI and local neural classifiers: an overview of the adaptive brain interface project," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on,* vol. 11, pp. 159-161, 2003.

[12] J. d. R. Millán, M. Franzé, J. Mouriño, F. Cincotti, and F. Babiloni, "Relevant EEG features for the classification of spontaneous motor-related tasks," *Biol. Cybern.,* vol. 86, pp. 89-95, 2002.

[13] K. LaFleur, K. Cassady, A. Doud, K. Shades, E. Rogin, and B. He, "Quadcopter control in three-dimensional space using a noninvasive motor imagery-based brain–computer interface," *J. Neural Eng.,* vol. 10, p. 046003, 2013.

[14] B. Blankertz, K.-R. Müller, D. J. Krusienski, G. Schalk, J. R. Wolpaw, A. Schlögl*, et al.*, "The BCI competition III: Validating alternative approaches to actual BCI problems," *IEEE T. Neur. Sys. Reh.,* vol. 14, pp. 153-159, 2006.

[15] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner*, et al.*, "Review of the BCI competition IV," *Front. Neurosci.,* vol. 6, 2012.

[16] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain–computer interfaces for communication and control," *Clin. Neurophysiol.,* vol. 113, pp. 767-791, 2002.

[17] C. Vidaurre, C. Sannelli, K.-R. Müller, and B. Blankertz, "Co-adaptive calibration to improve BCI efficiency," *J. Neural Eng.,* vol. 8, p. 025009, 2011.

[18] C. Vidaurre, M. Kawanabe, P. von Bünau, B. Blankertz, and K.-R. Müller, "Toward unsupervised adaptation of LDA for brain–computer interfaces," *IEEE T. Bio-med Eng.,* vol. 58, pp. 587-597, 2011.

[19] J. Andreu, R. D. Baruah, and P. Angelov, "Real time recognition of human activities from wearable sensors by evolving classifiers," in *Fuzzy Systems (FUZZ), 2011 IEEE International Conference on*, 2011, pp. 2786-2793.

[20] P. Herman, G. Prasad, and T. M. McGinnity, "Designing a robust type-2 fuzzy logic classifier for non-stationary systems with application in brain-computer interfacing," in *Systems, Man and Cybernetics, 2008. SMC 2008. IEEE International Conference on*, 2008, pp. 1343-1349.

[21] P. Herman, G. Prasad, and T. M. McGinnity, "Design and on-line evaluation of type-2 fuzzy logic system-based framework for handling uncertainties in BCI classification," in *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2008, pp. 4242-4245.

[22] A. Das, S. Suresh, N. Sundararajan, and K. Subramanian, "A subject-specific frequency band selection for efficient BCI-an interval type-2 fuzzy inference system approach," in *Proceedings of the IEEE International Conference on Fuzzy Systems*, 2015, pp. 1-8.

[23] H. Ramoser, J. Muller-Gerking, and G. Pfurtscheller, "Optimal spatial filtering of single trial EEG during imagined hand movement," *IEEE T. Rehabil. Eng.,* vol. 8, pp. 441-446, 2000.

[24] I. Gath and A. B. Geva, "Unsupervised optimal fuzzy clustering," *IEEE T. Pattern. Anal.,* vol. 11, pp. 773-780, 1989.

[25] D. Gustafson and W. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," in *IEEE Conference on Decision and Control*, 1978, pp. 1-6.

[26] N. R. Pal and J. C. Bezdek, "On cluster validity for the fuzzy c-means model," *IEEE T Fuzzy Syst,* vol. 3, pp. 370-379, 1995.

[27] F. Klawonn and F. Höppner, "What is fuzzy about fuzzy clustering? Understanding and improving the concept of the fuzzifier," in *Advances in Intelligent Data Analysis V*, ed: Springer, 2003, pp. 254-264.

[28] V. Schwämmle and O. N. Jensen, "A simple and fast method to determine the parameters for fuzzy c–means cluster analysis," *Bioinformatics,* vol. 26, pp. 2841-2848, 2010.

[29] M. Huang, Z. Xia, H. Wang, Q. Zeng, and Q. Wang, "The range of the value for the fuzzifier of the fuzzy c-means algorithm," *Pattern Recogn. Lett.,* vol. 33, pp. 2280-2284, 2012.

[30] J. Abonyi and F. Szeifert, "Supervised fuzzy clustering for the identification of fuzzy classifiers," *Pattern Recogn. Lett.,* vol. 24, pp. 2195-2207, 2003.

[31] J. M. Mendel and R. I. B. John, "Type-2 fuzzy sets made simple," *IEEE T Fuzzy Syst,* vol. 10, pp. 117-127, 2002.

[32] J. Mendel, H. Hagras, W.-W. Tan, W. W. Melek, and H. Ying, *Introduction To Type-2 Fuzzy Logic Control: Theory and Applications*: Wiley-IEEE Press, 2014.

[33] C. Wagner and H. Hagras, "Toward general type-2 fuzzy logic systems based on zSlices," *IEEE T Fuzzy Syst,* vol. 18, pp. 637-660, 2010.

[34] D. Wu and M. Nie, "Comparison and practical implementation of type-reduction algorithms for type-2 fuzzy sets and systems," Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ), 2011, pp. 2131-2138.

[35] N. N. Karnik and J. M. Mendel, "Centroid of a type-2 fuzzy set," *Inform. Sciences,* vol. 132, pp. 195-220, 2001.

[36] D. Wu, "An overview of alternative type-reduction approaches for reducing the computational cost of interval type-2 fuzzy logic controllers," in *IEEE International Conference on Fuzzy Systems*, 2012, pp. 1-8.

[37] N. V. Quoc, J.-H. Park, and S.-B. Choi, "Design of a novel adaptive fuzzy sliding mode controller and application for vibration control of magnetorheological mount," *Proc. Inst. Mech. Eng.,* vol. 228, pp. 2285-2302, 2014.

[38] M. Nie and W.-W. Tan, "Towards an efficient type-reduction method for interval type-2 fuzzy logic systems," in *Proceedings of the IEEE International Conference on Fuzzy Systems* 2008, pp. 1425-1432.

[39] P. Sanders, J. Speck, and R. Steffen, "Work-efficient matrix inversion in polylogarithmic time," *ACM Trans. on Parallel Computing,* vol. 2, p. 15, 2015.

[40] N.-Y. Liang, G.-B. Huang, P. Saratchandran, and N. Sundararajan, "A fast and accurate online sequential learning algorithm for feedforward networks," *IEEE T. Neural Networ.,* vol. 17, pp. 1411-1423, 2006.

[41] M. A. Woodbury, "Inverting modified matrices," *Memorandum report,* vol. 42, p. 106, 1950.

[42] O. Georgieva and D. Filev, "Gustafson-kessel algorithm for evolving data stream clustering," in *Proceedings of the International Conference on*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2016.2637403, IEEE Transactions on Fuzzy Systems

14

Computer Systems and Technologies and Workshop for PhD Students in Computing, 2009, p. 62.

[43] J. H. Zar, "Approximations for the percentage points of the chi-squared distribution," *Appl. Stat.,* pp. 280-290, 1978.

[44] E. Welzl, *Smallest enclosing disks (balls and ellipsoids)*: Springer, 1991.

[45] D. Ververidis and C. Kotropoulos, "Gaussian mixture modeling by exploiting the Mahalanobis distance," *IEEE T Signal Proces,* vol. 56, pp. 2797-2811, 2008.

[46] T. Kailath, "The divergence and Bhattacharyya distance measures in signal selection," *IEEE T. Commun.,* vol. 15, pp. 52-60, 1967.

[47] E. Lughofer and M. Sayed-Mouchaweh, "Autonomous data stream clustering implementing split-and-merge concepts–towards a plug-and-play approach," *Inform. Sciences,* vol. 304, pp. 54-79, 2015.

[48] K. J. Cios, W. Pedrycz, and R. W. Swiniarski, *Data mining methods for knowledge discovery* vol. 458: Springer Science & Business Media, 2012.

[49] E. Lughofer, C. Cernuda, S. Kindermann, and M. Pratama, "Generalized smart evolving fuzzy systems," *Evolving Systems,* vol. 6, pp. 269-292, 2015.

[50] I. Daly, R. Scherer, M. Billinger, and G. Müller-Putz, "FORCe: Fully Online and automated artifact Removal for brain-Computer interfacing," *IEEE T. Neur. Sys. Reh.,* vol. 23, pp. 725-736, 2015.

[51] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Adv. in large margin classifiers,* vol. 10, pp. 61-74, 1999.

[52] F. Lotte and C. Guan, "Regularizing common spatial patterns to improve BCI designs: unified theory and new algorithms," *IEEE T. Bio-med Eng.,* vol. 58, pp. 355-362, 2011.

[53] F. Babiloni, F. Cincotti, L. Bianchi, G. Pirri, J. d. R. Millán, J. Mouriño*, et al.*, "Recognition of imagined hand movements with low resolution surface Laplacian and linear classifiers," *Med. Eng. Phys.,* vol. 23, pp. 323-328, 2001.

[54] C. Brunner, R. Scherer, B. Graimann, G. Supp, and G. Pfurtscheller, "Online control of a brain-computer interface using phase synchronization," *IEEE T. Bio-med Eng.,* vol. 53, pp. 2501-2506, 2006.

[55] J. Faller, C. Vidaurre, T. Solis-Escalante, C. Neuper, and R. Scherer, "Autocalibration and recurrent adaptation: towards a plug and play online ERD-BCI," *IEEE T. Neur. Sys. Reh.,* vol. 20, pp. 313-319, 2012.

[56] R. D. Baruah, P. Angelov, and J. Andreu, "Simpl_eClass: simplified potential-free evolving fuzzy rule-based classifiers," in *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on*, 2011, pp. 2249-2254.

[57] J. Andreu and P. Angelov, "An evolving machine learning method for human activity recognition systems," *Journal of Ambient Intelligence and Humanized Computing,* vol. 4, pp. 195-206, 2013.

[58] G. E. Hinton, "Deep belief networks," *Scholarpedia,* vol. 4, p. 5947, 2009.

[59] B. Cetişli and A. Barkana, "Speeding up the scaled conjugate gradient algorithm and its application in neuro-fuzzy classifier training," *Soft Comput.,* vol. 14, pp. 365-378, 2010.

[60] W.-Y. Hsu, "EEG-based motor imagery classification using neuro-fuzzy prediction and wavelet fractal features," *J. Neurosci. Meth.,* vol. 189, pp. 295-302, 2010.

[61] R. Palaniappan, R. Paramesran, S. Nishida, and N. Saiwaki, "A new brain-computer interface design using fuzzy ARTMAP," *IEEE T. Neur. Sys. Reh.,* vol. 10, pp. 140-148, 2002.

[62] D. Steyrl, R. Scherer, and G. Müller-Putz, "Using random forests for classifying motor imagery EEG," in *Proceedings of TOBI workshop IV*, 2013, pp. 89-90.

[63] S. M. Hosni, M. E. Gadallah, S. F. Bahgat, and M. S. AbdelWahab, "Classification of EEG signals using different feature extraction techniques for mental-task BCI," in *Proceedings of the International Conference onComputer Engineering & Systems (ICCES)*, 2007, pp. 220-226.

[64] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for EEG-based brain–computer interfaces," *J. Neural Eng.,* vol. 4, p. R1, 2007.

[65] M. G. Rosenblum, A. S. Pikovsky, and J. Kurths, "Phase synchronization of chaotic oscillators," *Phys. Rev. Lett.,* vol. 76, p. 1804, 1996.

Javier Andreu-Perez (S'10-M'13) is Research Associate at the Hamlyn Centre, within the Department of Computing of Imperial College London, U.K, where he is a member of the Machine Learning and Human Robot Interaction group. He holds a PhD in Intelligent Systems, a MSc in Software Engineering, and a MEng in Computer Science and Engineering. He is a member of the Standards Committee of the IEEE Society in Computational Intelligence. He also serves as associate editor in several relevant journals of computational intelligence. His research interests include computational intelligence, fuzzy systems, neural networks, biomedical engineering, health informatics, neurotechnology.
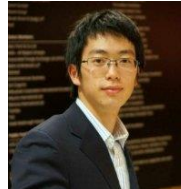
Fan Cao obtained his MRes in Medical Robotics and Image Guided Intervention from the Hamlyn Centre of Imperial College London, U.K. His main research interests are neurocience and brain computer interfaces. He is currently working at g.tec medical engineering, Austria.

Hani Hagras (M'03–SM'05-F'13) received the B.Sc. and M.Sc. degrees in electrical engineering from Alexandria University, Alexandria, Egypt, and the Ph.D. degree in computer science from the University of Essex, Colchester, U.K. He is a Professor in the School of Computer Science and Electronic Engineering, Director of the Computational Intelligence Centre and the Head of the Fuzzy Systems Research Group in the University of Essex. His major research interests are in computational intelligence, notably type-2 fuzzy systems, fuzzy logic, neural networks, genetic algorithms, and evolutionary computation. He is also interested in ambient intelligence, pervasive computing and intelligent buildings, as well as embedded agents, robotics and intelligent control. He is a Fellow of the Institute of Electrical and Electronics Engineers (IEEE) and a Fellow of the Institution of Engineering and Technology (IET). His research has won numerous prestigious international awards, including most recently from the IEEE Computational Intelligence Society (CIS). He is also Associate Editor of several prestigious journals in the field of computational intelligence and robotics, and is a member of the IEEE CIS Fuzzy Systems Technical Committee and IEEE CIS Conference Committee.

Guang-Zhong Yang (S'90–M'91–SM'08–F'11) received his PhD in Computer Science from Imperial College London, U.K. He is the director, co-founder of the Hamlyn Centre for Robotic Surgery, Deputy Chairman of the Institute of Global Health Innovation. Professor Yang also holds a number of key academic positions at Imperial College – he is director and founder of the Royal Society/Wolfson Medical Image Computing Laboratory, co-founder of the Wolfson Surgical Technology Laboratory, and chairman of the Centre for Pervasive Sensing. He is also a Distinguished Lecturer for IEEE Engineering in Medicine and Biology Society (EMBS) and a Fellow of the Royal Academy of Engineering (RAEng), the Institute of Electrical and Electronics Engineers (IEEE), the Institution of Engineering and Technology (IET), the American Institute for Medical and Biological Engineering (AIMBE), the International Academy of Medical and Biological Engineering (IAMBE), the Society of Medical Imaging and Computer Assisted Intervention (MICCAI), and the City and Guilds of London. He is equally a recipient of the Royal Society Research Merit Award, the I.I. Rabi Award from the International Society for Magnetic Resonance in Medicine and The Times Eureka 'Top 100' in British Science. He is also editor-in-chief of prestigious journals in the fields of health informatics and robotics. He pioneered the concept of perceptual docking for robotic control as well as the field of Body Sensor Networks (BSN). Professor Yang's main research interests are in medical imaging, sensing and robotics.