

Interface cerveau machine avec adaptation à l'utilisateur

Xavier ARTUSI¹, Imran KHAN NIAZI², Marie-Françoise LUCAS¹, Darion FARINA²

¹Institut de Recherche en Communication et Cybernétique de Nantes (IRCCyN)
Centrale Nantes, 1 rue de la Noë, 44321 Nantes, France

²Center for Sensory-Motor Interaction Department of Health Science and Technology,
Aalborg University, Aalborg, Denmark

xavier.artusi@irccyn.ec-nantes.fr, imrankn@hst.aau.dk
marie-francoise.lucas@irccyn.ec-nantes.fr, dario.farina@bccn.uni-goettingen.de

Résumé – L'objet de cet article est l'étude d'une interface cerveau-machine (BCI) corrigée par la prise en compte d'une réaction involontaire du sujet lors de l'affichage de la sortie de la BCI et pouvant s'adapter aux évolutions de l'utilisateur. L'apport de ces travaux est d'une part l'étude théorique des performances du système corrigé et d'autre part la simulation du système adaptatif. Les résultats obtenus sur des données EEG réelles montrent que l'utilisation du détecteur d'erreur augmente significativement les performances de l'interface, et l'étude en simulation montre l'adaptativité du système pour une stratégie donnée de mise à jour de l'apprentissage.

Abstract – The purpose of this paper is the study of a brain-computer interface (BCI) corrected using an involuntary reaction of the user in response to an error made by the translation algorithm. First, we theoretically quantify the improvement in accuracy of the corrected BCI system. Second, we study in simulation its ability to adapt to the changes in the mental states of the user. Results on real EEG data show that the use of error detector significantly increases the performance of the interface and simulations show the adaptivity of the system for a given strategy to update the learning set.

1 Introduction

Une interface cerveau-machine (BCI, Brain Computer Interface) est un système qui permet la communication directe entre le cerveau d'un individu et un ordinateur ou un robot, sans solliciter les nerfs périphériques et les muscles. Le but est de permettre à des personnes souffrant d'un handicap neuromusculaire sévère de communiquer avec leur entourage ou de commander une prothèse par la pensée. Nous nous intéressons ici à des systèmes permettant de commander une prothèse. L'entrée du système est constituée de signaux électroencéphalographiques (EEG) enregistrés à la surface du scalp à l'aplomb des aires sensorimotrices du cortex et reliés à l'activité volontaire du sujet. La sortie est une décision se traduisant par une action de commande d'une prothèse pour effectuer un mouvement particulier parmi n possibles. Une période d'apprentissage est nécessaire pour construire les fonctions de décision permettant l'exécution des actions correspondant aux intentions du sujet.

Dès lors qu'il s'agit de faire des distinctions complexes entre des tâches imaginaires (ici différentes modalités d'un même mouvement), les BCI classiques ne sont pas robustes, nécessitent des sessions d'entraînement fastidieuses et ne s'adaptent pas. Or au cours du temps, le sujet évolue, car ayant connaissance des réponses du système, il s'y adapte en affinant sa stratégie de commande (biofeedback). Le contexte de ce travail est de développer un système BCI capable d'améliorer ses performances en cours d'utilisation et de s'adapter à l'utilisateur sans

nécessiter de multiples sessions d'apprentissage.

Une approche pour y parvenir est d'inclure un retour de l'utilisateur sur le système de décision [1, 2, 3] : l'idée est d'estimer l'erreur de la BCI à partir de potentiels cérébraux appelés potentiels d'erreur, évoqués chez le sujet et corrélés à l'échec de la décision prise par l'interface. La détection des potentiels d'erreur sera utilisée ici pour améliorer les performances de la BCI et la rendre adaptative. L'objectif de cet article est double : il s'agit d'une part de quantifier l'amélioration des performances de l'interface BCI corrigée par la prise en compte des potentiels d'erreur, et d'autre part d'étudier en simulation l'adaptation du système corrigé aux changements de l'utilisateur.

2 Présentation du système adaptatif

Le système représenté sur la Figure 1 comporte 2 modules d'analyse et décision de signaux EEG. L'objet du module principal (noté Système de décision) est le décodage de l'intention de l'utilisateur : il produit et affiche une décision de mouvement à partir des signaux EEG_1 images de l'intention de mouvements. Le module secondaire (noté Détecteur d'erreur) estime la présence ou l'absence d'une erreur sur la décision précédente en analysant les signaux EEG_2 enregistrés après l'affichage de la décision et supposés contenir une information pertinente sur la justesse de cette décision. Sa sortie est utilisée pour corriger les décisions du module principal, et mettre à jour la base d'ap-

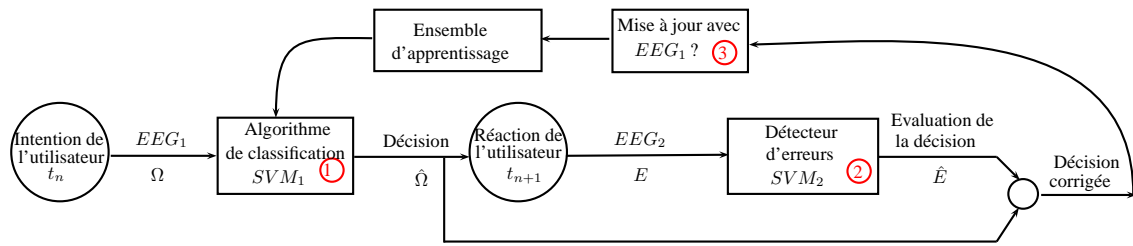


FIGURE 1 – Schéma du système adaptatif

prentissage avec des EEG_1 pertinents. La mise en oeuvre de ce schéma nécessite de disposer d'un système de décision pouvant apprendre en ligne, un dispositif détectant ses erreurs, et une stratégie de mise à jour de sa population d'apprentissage.

- **Système de décision :** Cette étape consiste à décoder les intentions de l'utilisateur à partir des EEG_1 (cortex moteur), ici différentes vitesses d'exécution d'un même mouvement (flexion *lente/rapide* du bras droit). La discrimination de tels mouvements se fait grâce aux potentiels corticaux liés au mouvement (MRCPs) dont l'information se situe dans les basses fréquences. Comme proposé dans [4], nous utilisons comme descripteurs les marginales de la décomposition en ondelettes (DWT), qui représentent l'intensité moyenne du signal sur des bandes de fréquence selon un découpage dyadique.
- **Détecteur d'erreur :** La détection des erreurs du système se fait grâce au signal EEG_2 , par classification supervisée, en considérant les classes *juste/faux* (absence/présence d'un potentiel d'erreur). [1] a montré qu'un signal EEG spécifique (Potentiel d'erreur, ErrP) apparaît au niveau du cortex frontal entre 200ms et 700ms après que la BCI ait affiché une réponse fautive. Les descripteurs utilisés pour la classification sont les échantillons temporels entre 150ms et 650ms obtenus après filtrage (0 – 10Hz) et sous-échantillonnage (32 descripteurs par voie, 2 voies) [5].

Des SVM linéaires (SVM_1 fonctionnant en ligne [6] et SVM_2), appris sur des populations d'apprentissage, sont utilisés pour classer les signaux dans les deux cas. Cette méthode est considérée comme robuste même dans des conditions difficiles (dimension de l'espace de représentation grande par rapport à la taille de l'ensemble d'apprentissage). Si le nombre de classes de mouvement n est supérieur à 2, on utilise une procédure de one-versus-rest (OVR) avec n SVM linéaires, chacun séparant une classe du reste de la population. Les performances du système de décision SVM_1 corrigé par le détecteur SVM_2 sont déterminées théoriquement en Section 3 et sur la base de données expérimentales en Section 4.

- **Mise à jour de sa population d'apprentissage :** Pour adapter le BCI aux changements de l'utilisateur, l'ensemble d'apprentissage doit être mis à jour avec des exemples pertinents et le SVM_1 doit modifier en ligne la fonction de décision en tenant compte de ces changements. Pour cela, il est nécessaire de définir une stratégie visant à mo-

difier l'ensemble d'apprentissage. Comme l'adaptation est l'objectif, nous utilisons une fenêtre glissante : quand un nouvel exemple est intégré à l'ensemble d'apprentissage, l'exemple le plus ancien est supprimé. Pour intégrer un nouvel élément, il est nécessaire d'avoir une bonne estimation de sa classe. Ceci est vrai uniquement pour les exemples qui sont considérés corrects par le détecteur d'erreur. Deux stratégies sont testées pour mettre à jour l'ensemble d'apprentissage :

- prendre tous les exemples considérés corrects par le détecteur,
- parmi les exemples considérés corrects par le détecteur, ne prendre que les exemples proches de la surface de décision, parce qu'ils sont considérés comme plus informatifs. Cela correspond à la technique d'échantillonnage par incertitude utilisée en apprentissage actif. Cette proximité est mesurée par la valeur de la fonction de décision f (ici x est intégré si $|f(x)| > 1$) [7].

Nous avons développé un simulateur pour comparer ces deux stratégies entre elles et à une référence connue (Section 5).

3 Performances théoriques en sortie du système corrigé

Le détecteur renvoie une information sur la justesse de la réponse du SVM, mais non sur la classe. Si la réponse est détectée comme *faux* dans le cas général multiclassé, il n'est pas possible de déduire la vraie classe même avec un détecteur parfait. Ainsi une stratégie naturelle pour prendre en compte la réponse du détecteur est la suivante :

- lorsque la réponse du SVM_1 est estimée *faux*, le sujet doit répéter la tâche.
- sinon, on active la prothèse en accord avec la décision du SVM_1 .

Le problème est formalisé en utilisant quatre variables aléatoires (Figure 1) :

- Ω et $\hat{\Omega}$ à valeurs dans $\{\omega_1, \dots, \omega_n\}$ représentent respectivement l'intention de mouvement et l'intention décodée par le SVM_1 (avec n le nombre de classes), $\omega, \hat{\omega}$ étant les réalisations correspondantes ;
- E et \hat{E} à valeurs dans $\{juste, faux\}$ représentent la justesse de la décision du SVM et son estimation par le détecteur, e, \hat{e} étant les réalisations correspondantes ; $e = juste$

si $\hat{\omega} = \omega$ sinon $e = faux$.

En posant :

- $P_{\Omega}(\omega) = \text{Prob}(\Omega = \omega)$: probabilité *a priori* de la classe ω ,
- $P_{\hat{\Omega}|\Omega}(\hat{\omega}, \omega) = \text{Prob}(\hat{\Omega} = \hat{\omega} | \Omega = \omega) = P_E(e)$: densité de la réponse du SVM_1 conditionnellement à l'intention de mouvement,
- $P_{\hat{E}|\hat{\Omega}\Omega}(\hat{e}, \hat{\omega}, \omega) = P_{\hat{E}|E}(\hat{e}, e) = \text{Prob}(\hat{E} = \hat{e} | E = e)$: densité de la réponse du détecteur SVM_2 conditionnellement à la justesse de la réponse du SVM_1 .

La probabilité d'erreur de commande en sortie du système corrigé (P_{err}) correspond à la probabilité que, lorsque le détecteur estime la décision du SVM_1 *juste*, l'intention décodée par le SVM_1 soit différente de l'intention de l'utilisateur :

$$P_{Err} = \sum_{\substack{i,j \\ i \neq j}} P_{\Omega\hat{\Omega}|\hat{E}}(\omega_i, \hat{\omega}_j, \text{juste}) \\ = \sum_{\substack{i,j \\ i \neq j}} \frac{P_{\hat{E}|\hat{\Omega}\Omega}(\hat{e}, \hat{\omega}_j, \omega_i) \cdot P_{\hat{\Omega}|\Omega}(\hat{\omega}_j, \omega_i) \cdot P_{\Omega}(\omega_i)}{P_{\hat{E}}(\text{juste})} \quad (1)$$

avec $P_{\hat{E}}(\text{juste}) = \sum_e P_{\hat{E}|E}(\text{juste}, e) \cdot P_E(e)$.

La performance du système est également caractérisée par le taux de répétition, P_r qui correspond à la probabilité que le détecteur donne la réponse *faux*, $P_{\hat{E}}(faux)$.

4 Résultats sur des données EEG réelles

Bien que le système adaptatif ne soit pas testé en ligne, les modules de classification de mouvements et de détection d'erreur ont été testés hors ligne sur des signaux EEG réels produits selon un protocole adapté au schéma de la figure 1. Les résultats de la classification sont présentés Tableau 1. Le taux d'erreur du SVM_1 , similaire pour les deux classes (*lent/rapide*) est de 26% en moyenne sur tous les sujets avec un écart type faible (4%). Ces résultats sont utilisés comme estimation des probabilités conditionnelles $P_{\hat{\Omega}|\Omega}$ du SVM_1 . Dans le Tableau 2 nous présentons les résultats de la classification des ErrP, qui sont utilisés comme estimées des probabilités conditionnelles du détecteur $P_{\hat{E}|E}$. A partir de l'équation (1) et des résultats précédents nous avons calculé le taux d'erreur de la BCI corrigée ainsi que la probabilité de répétition (Tableau 3). On constate qu'en moyenne le taux d'erreur est pratiquement divisé par deux, pour un taux de répétition de 20% (1 fois sur 5). Ces résultats montrent que malgré une reconnaissance imparfaite des potentiels d'erreur, l'utilisation du détecteur augmente significativement les performances de la BCI.

5 Simulation de l'ensemble du système bouclé

Le simulateur présenté dans ce paragraphe permet de tester l'adaptativité du système avec différentes stratégies de mises à

Sujets	1	2	3	4	5	6	Moy.
Taux d'erreur	28%	24%	32%	30%	22%	22%	26%

TABLE 1 – Pourcentage de mal classés des signaux EEG_1 .

Sujets	1	2	3	4	5	6	Moy.
classe <i>juste</i>	92%	91%	93%	87%	83%	80%	88%
classe <i>faux</i>	73%	43%	60%	60%	37%	63%	56%

TABLE 2 – Pourcentage de bien classés des signaux EEG_2 .

Sujet	1	2	3	4	5	6	Moy.
P_{Err}	10%	16%	17%	17%	18%	11%	14%
P_r	26%	17%	23%	27%	21%	30%	20%

TABLE 3 – Taux d'erreur du système corrigé (à comparer au taux d'erreur du SVM_1 seul, Tableau 1) et probabilité de répétition.

jour de l'apprentissage. Cette étude ne peut se faire qu'en simulation, car il faut estimer à chaque instant la performance du système évolutif (ce qui ne peut être effectué sur des données expérimentales à partir d'un essai) et la comparer à une référence connue. Ces tests sont réalisés en simulant non pas les signaux, mais directement des vecteurs de descripteurs dans un espace de représentation à deux dimensions : l'objectif n'est pas de simuler des EEG réalistes, mais des classes caractérisées par des densités de probabilité connues et telles que les performances du classifieur soient du même ordre que celles obtenues dans le cas expérimental. On considère ainsi dans \mathbb{R}^2 n classes $\{\omega_1, \dots, \omega_i, \dots, \omega_n\}$. Les éléments de la classe ω_i (un élément simulant un essai) sont distribués selon une densité gaussienne $\mathcal{N}(\mu_i, \Sigma_i)$ qui peut évoluer. Les opinions \hat{e} du détecteur sont les réalisations d'une variable aléatoire binaire \hat{E} de densité conditionnelle $P_{\hat{E}|E}$.

Quand un nouvel exemple est généré, le classement est effectué en ligne par le SVM_1 multiclasse. Comme initialisation, l'ensemble d'apprentissage est composé de N_{init} exemples étiquetés. Ensuite, les nouveaux éléments sont ou non intégrés avec l'étiquette estimée par le SVM_1 , selon l'opinion \hat{e} du détecteur et la stratégie choisie. La taille de l'ensemble d'apprentissage augmente jusqu'à N_w correspondant à la taille de la fenêtre glissante.

A chaque instant, la probabilité d'erreur de classification est estimée empiriquement à partir d'un ensemble de test de 1000 exemples générés selon les densités courantes $\mathcal{N}(\mu_i, \sigma_i)$, dans quatre cas :

- le SVM_1 sans détecteur d'erreur : la décision du SVM_1 n'est pas corrigée et l'apprentissage est mis à jour en intégrant chaque nouvel exemple avec l'étiquette de la classe décodée par le SVM_1 ;
 - le classifieur bayésien idéal connaissant la densité théorique à tout instant, sans détecteur d'erreur pour corriger sa décision ;
 - le SVM_1 et le détecteur d'erreur : la décision du SVM_1 est corrigée et l'ensemble d'apprentissage mis à jour selon le détecteur ;
 - le classifieur bayésien idéal corrigé par le détecteur d'erreur.
- Les classifieurs bayésiens sont utilisés comme références opti-

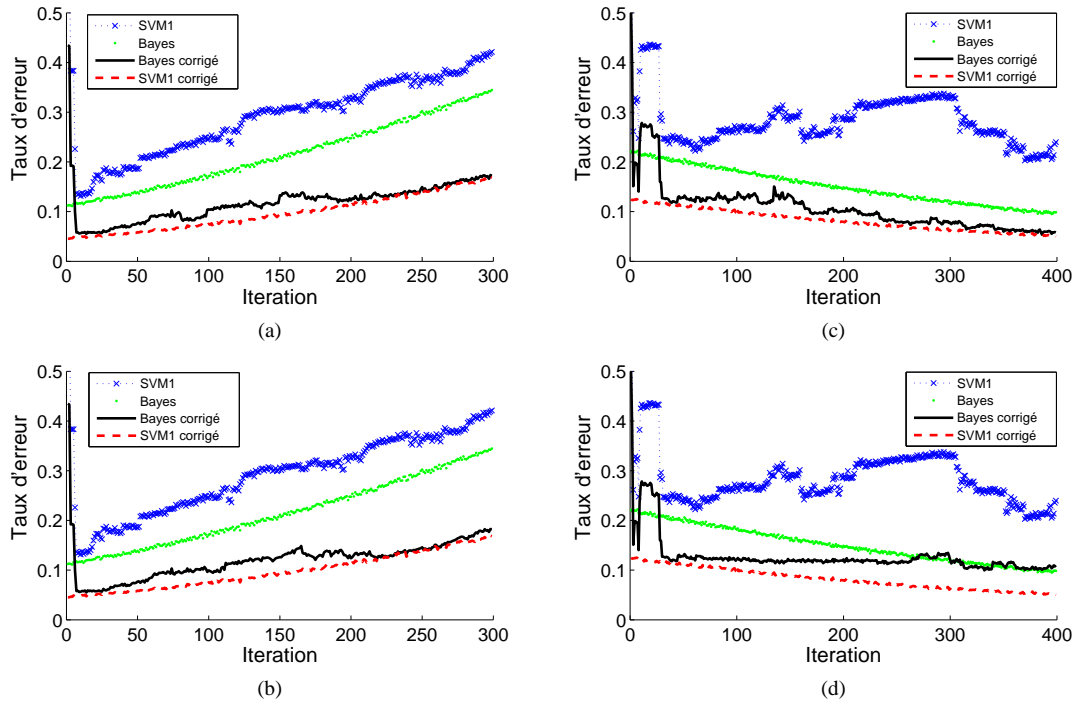


FIGURE 2 – Simulation du système adaptatif avec les classes translatant et la distance entre les centres diminuant (a,b) et avec la distance entre les centres augmentant sans translation des classes (c,d). Les stratégies utilisées pour mettre à jour l'ensemble d'apprentissage sont : (a,c) prendre tous les exemples estimés comme *juste* par le détecteur, (b,d) utiliser la technique d'échantillonnage par incertitude parmi ces exemples, prendre seulement les exemples proches de la séparatrice.

males pour les SVM_1 .

Nous présentons les résultats de deux simulations à 3 classes. Dans la première simulation les centres des classes se rapprochent, tandis que dans la seconde ils s'éloignent et se translatent. Les longueurs des fenêtres d'apprentissage initial et glissant sont respectivement $N_{init} = 10$ et $N_w = 40$. Les probabilités conditionnelles du détecteur correspondent aux valeurs moyennes obtenues sur les données expérimentales : $P_{\hat{E}|E}(juste, juste) = 88\%$ et $P_{\hat{E}|E}(faux, faux) = 56\%$. Dans le cas de la première simulation (Figures 2a and 2b), les deux stratégies convergent vers le classifieur bayésien idéal, alors que le SVM seul diverge. La seconde simulation (Figures 2c and 2d), où les classes sont translatées et la distance entre les centres augmente, montre que la première stratégie (prendre tous les exemples estimés comme *juste*) permet l'adaptation et la convergence du classifieur alors que la seconde stratégie (échantillonnage par incertitude) ne converge pas. Ces résultats montrent que le système bouclé peut s'adapter, et que la meilleure stratégie est celle qui consiste à prendre tous les exemples estimés *juste* par le détecteur.

Références

- [1] P.W. FERREZ et J.D.R. MILLÁN : Error-Related EEG Potentials Generated During Simulated Brain-Computer Interaction. *IEEE transactions on biomedical engineering*, 55(3):923–929, 2008.
- [2] P.W. FERREZ et J. MILLÁN : You Are Wrong Automatic Detection of Interaction Errors from Brain Waves. *International Joint Conference On Artificial Intelligence*, 19:1413–1418, 2005.
- [3] B. DAL SENO, M. MATTEUCCI et L. MAINARDI : Online Detection of P300 and Error Potentials in a BCI Speller. *Computational Intelligence and Neuroscience*, pages 1–5, 2010.
- [4] D. FARINA, O.F. NASCIMENTO, M.F. LUCAS et C. DONCARLI : Optimization of wavelets for classification of movement-related cortical potentials generated by variation of force-related parameters. *Journal of neuroscience methods*, 162(1-2):357–363, 2007.
- [5] A. RAKOTOMAMONJY, V. GUIGUE, G. MALLET et V. ALVARADO : Ensemble of SVMs for improving brain computer interface P300 speller performances. *Artificial Neural Networks : Biological Inspirations–ICANN 2005*, pages 45–50, 2005.
- [6] G. CAUWENBERGHS et T. POGGIO : Incremental and decremental support vector machine learning. *Proceedings of Advances in neural information processing systems*, 13:409–415, 2001.
- [7] G. SCHÖHN et D. COHN : Less is More : Active Learning with Support Vector Machines. *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 839–846, 2000.