

IDENTIFICATION PAR EEG ET UTILISATION DANS UN CADRE ARTISTIQUE DES ÉTATS DE CONCENTRATION : VERS UNE SONIFICATION DE L'ATTENTION ?

Armand Bazin
Université Gustave Eiffel
bazin.armand@gmail.com

Ulysse Del Ghingaro
Musidanse/C.et.I (EA1572)
Université Paris 8
ulyssedgo@gmail.com

RÉSUMÉ

Ce projet de recherche fait suite à un premier article publié aux JIM 2020, concluant sur certaines possibilités et limites des technologies d'électro-encéphalogrammes (EEG) dans un cadre de recherche et création. En reprenant une méthodologie proche, ce nouvel article contient une proposition d'un dispositif de sonification des données d'un casque EEG. L'objectif du projet est de développer un environnement sonore interactif en favorisant les rétroactions entre le dispositif de mesure et l'utilisateur, notamment à travers la sonification intuitive de ses états de concentration. Ces états de concentration sont discriminés par apprentissage automatique grâce à la bibliothèque Java *BrainFlow*.

L'article explicite la façon dont les données ont été mesurées et traitées ainsi que les choix de mapping et de contrôle de la création sonore. Le projet étant encore en développement, la partie discussion ouvre sur des pistes d'améliorations concernant la synthèse sonore et les méthodes de classification des états concentrés / relaxés.

1. INTRODUCTION

Ce projet mobilise d'un côté des connaissances d'analyse de données et d'apprentissage automatique et de l'autre une volonté d'intégrer à la composition musicale des réflexions écologiques – en pensant la musique comme le résultat d'interactions composées entre le porteur du casque et son environnement (sonore) à l'image des travaux du compositeur Agostino Di Scipio [1].

Ce qui nous intéresse particulièrement dans l'utilisation du casque EEG est la possibilité de créer une boucle de rétroaction entre l'auditeur, porteur du casque, et le son résultant de la sonification des données du casque. L'idée de départ de ce projet pouvait s'apparenter à une sonification de l'"écoute" de l'utilisateur. Cependant, identifier des réactions d'écoute par EEG s'est révélé trop complexe.

Nous avons donc décidé de nous concentrer sur la mesure de l'état de concentration de l'utilisateur, en cherchant à créer un environnement sonore capable

d'interagir avec sa capacité à se relaxer ou à se concentrer, via un retour sonore intuitif de l'état de concentration détecté par le système. Le dispositif est voulu simple : un environnement sonore, créé à partir d'une synthèse soustractive basée sur le signal reçu du casque, et modifié par le couple relaxation / concentration. Ainsi, un des buts de notre dispositif est de s'approcher d'une sonification de l'attention du porteur du casque.

Le terme de sonification recouvre une grande diversité de démarches pratiquées à la fois par des artistes sonores et des compositeurs. Une de ses définitions est de rendre sensible, grâce au son, certains phénomènes non perceptibles en temps normal [2].

Cette démarche est largement répandue dans les arts sonores écologiques qui cherchent, entre autres, à initier une écoute dite "écologique" chez les auditeurs, c'est-à-dire un mode d'écoute "dans lequel l'attention s'éloigne des choses ou des objets isolés au profit d'une dynamique plus globale centrée sur celle des interactions et des interrelations." [3].

Pour donner deux exemples on peut citer l'artiste Andrea Polli et son projet *Atmospheric weathers works* [4], basé sur de la sonification de données météorologiques ou bien *Le souffle de la Terre* de l'artiste Lorella Abenavoli [5] qui rend audible des données sismiques.

Dans ce projet de recherche, la technique de l'EEG est donc appréhendée comme une outil pour rendre concrètes et sensibles les interactions cognitives entre un auditeur et son environnement sonore selon l'attention qu'il y porte. Cette approche de la composition provient aussi de l'intégration de réflexions tirées de l'écologie de l'attention, élaborée par Yves Citton [6], et de l'écologie de l'esprit, conceptualisée par Gregory Bateson [7]. Ces réflexions ne seront pas développées dans cet article pour nous concentrer exclusivement sur l'aspect technique et sonore du dispositif.

2. MATÉRIEL ET MÉTHODE

2.1. Méthode

Ce travail s'intègre dans une démarche de recherche et création. Étant non spécialistes, notre approche consiste à essayer de s'approprier artistiquement les possibilités d'un casque électro-encéphalogramme (EEG), tout en essayant de rester au plus proche de la signification des données captées.

Cet article provient de la suite du travail qui avait résulté en une précédente publication [8]. Cette première version avait présenté des tests du casque *Open_BCI* mark IV ainsi qu'un projet de création appelé CAANEVA (création autonome par analyse neuronale d'environnement visuel et auditif).

Le présent article repart des conclusions du précédent pour mettre à jour les avancées dans le domaine des EEG par la firme open source *Open_BCI*¹ et propose un nouveau projet créatif plus en adéquation avec les capacités de ce matériel. Ce nouveau travail se base donc sur les objectifs suivants : simplifier le dispositif aussi bien au niveau matériel – un nombre réduit d'électrode avec une meilleure méthode d'analyse des données – qu'au niveau créatif – une boucle de rétroaction (feedback) sonore simple à appréhender – dans une volonté de permettre plus d'interactivité et de contrôle à l'utilisateur.

Des essais ont été menés avec le casque *OpenBCI* pour identifier la méthode de classification la plus appropriée et les activités correspondantes à l'état "relaxation" et l'état "concentration". Des modifications ont été apportées au code source du logiciel *OpenBCI GUI* afin de récupérer certaines données. Ces données ainsi qu'une Transformée de Fourier Rapide (FFT²) des fréquences d'activation neuronale sont les métriques sur lesquelles la création sonore est basée. Celle-ci est assurée par le logiciel *Max/MSP*.

Nous nous concentrerons ici sur les changements entre la première version et la seconde version du dispositif, le matériel n'étant pas tout à fait le même³.

2.2. Hardware et tests

Au sein de ce projet nous avons utilisé un casque *OpenBCI mark IV* huit canaux avec électrodes sèches de 5 mm. L'objectif étant de mesurer l'état de relaxation ou de concentration de l'utilisateur nous avons placé les électrodes selon la disposition 10-20 en Fp1, Fp2, F3 et F4. "10-20" fait référence aux rapports de distance entre deux électrodes adjacentes qui représente 10% ou 20 % de la longueur et largeur totale du crâne [9].

Les électrodes servant de terre et de référence sont positionnées sur les lobes d'oreille du porteur du casque en A1 et A2 (Figure 1).

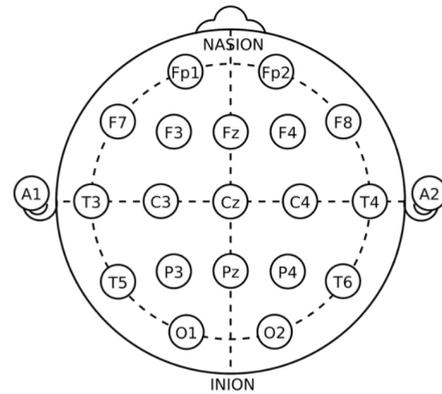


Figure 1. Disposition 10-20.

Les électrodes en F3 et F4 ne sont pas obligatoires pour les mesures mais peuvent servir de données complémentaires. Spécifiquement pour ces électrodes, nous avons rajouté des têtes d'électrodes "dry EEG comb electrodes"⁴ qui permettent d'obtenir un signal convenable malgré la chevelure.

Afin de récupérer les données des électrodes nous avons utilisé une carte d'*OpenBCI* à 4 canaux appelée *Ganglion*. Pour son utilisation en EEG, le pin de chaque canal a été réglé sur *down* comme indiqué dans la documentation *OpenBCI*⁵, passant les électrodes en configuration unipolaire⁶.

La récupération des données par l'ordinateur se fait grâce à une clé USB bluetooth et au logiciel *OpenBCI GUI*.

Les tests du hardware ont été très similaires aux tests menés dans le cadre du premier article, avec l'utilisation du widget de test d'impédance en plus.

Le widget de test d'impédance du ganglion est un outil dont l'utilisation est indispensable avant chaque session. En envoyant un courant dans chaque électrode de façon successive, il permet de calculer leur l'impédance. Cela permet de vérifier si ces dernières sont bien en contact avec la peau et donc si les données qu'elles renvoient ne sont pas des artefacts. Une impédance en dessous de 50 kΩ indique un bon contact de l'électrode. Réussir à réduire l'impédance des électrodes sous ce seuil n'est pas simple du fait de leur très grande sensibilité à tout changement de position.

Pour les électrodes placées en F3 et F4, les *comb electrodes* semblent bien fonctionner car une faible impédance est obtenue rapidement une fois les électrodes en contact avec le cuir chevelu.

Pour les électrodes placées en Fp1 et Fp2, l'opération est plus délicate. Nous avons testé différentes têtes d'électrodes sans changement notable. La position et le

1 <https://openbci.com/>

2 Fast Fourier Transform en anglais

3 Des informations plus approfondies sur les bases de l'EEG sont disponible dans l'article de 2020 [8]

4 <https://shop.openbci.com/products/5-mm-spike-electrode-pack-of-30>

5 <https://docs.openbci.com/AddOns/Headwear/HeadBand/#headband-ganglion-tutorial>

6 https://www.neuro.mcw.edu/meg/index.php/EEG_reference_with_Brainstorm

serrage des électrodes sont à modifier subtilement jusqu'à ce que l'impédance baisse convenablement.

Quant aux références, pour chaque session il nous a fallu attendre environ 5 minutes sans toucher au positionnement des électrodes de référence le temps que leur impédance soit bonne. Ce changement d'impédance peut être dû à la sudation qui améliore la conductivité entre la peau et l'électrode.

Enfin, nous avons pris soin de rassembler les câbles d'électrodes et de bien les fixer au casque pour limiter au maximum leurs mouvements, qui sont source d'artéfacts.

Ces remarques peuvent paraître superflues, mais dans le cadre d'une présentation artistique il est important de noter que ce matériel nécessite un temps de mise en place et de subtils ajustements avant de pouvoir l'utiliser correctement (Figure 2).

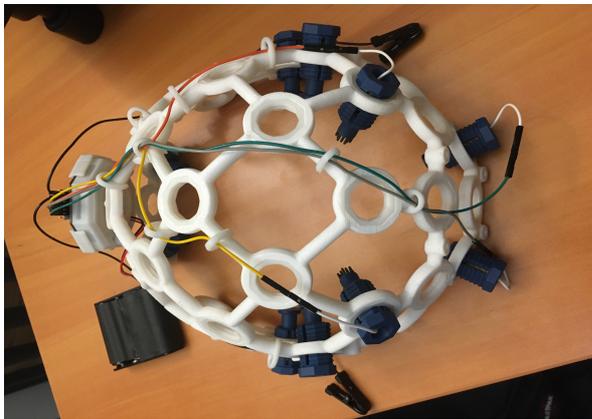


Figure 2. Casque *OpenBCI* Mark IV.

2.3. *OpenBCI* GUI et *Focus Widget*

Pour récupérer les données du ganglion nous avons utilisé le logiciel à interface graphique *OpenBCI GUI* (version 5.1.0).

OpenBCI GUI est un logiciel libre codé sous *Processing* (Java). Le fait qu'il soit open source nous a permis d'accéder à son code et de le modifier afin de récupérer une variable initialement non exportée.

Cette variable, appelée *metric value*, correspond au taux de confiance des différents classifieurs⁷ utilisés dans la détermination de l'état de concentration de l'utilisateur lors du port du système, à savoir concentré, neutre ou relaxé.

Ce "taux de confiance" établit la probabilité avec laquelle le classifieur estime l'état mental de l'utilisateur. Elle oscille ainsi entre 0 et 1 pour chacun des deux états, avec une relation inversement proportionnelle telle que :

$$P(\text{Etat concentré}) \approx 1 - P(\text{État Relaxé}) \quad (1)$$

⁷ Les classifieurs sont des algorithmes d'apprentissage automatique, ici en l'occurrence d'apprentissage supervisé ayant pour but de déterminer la catégorie d'appartenance de données. Dans notre cas, les classifieurs sont utilisés pour classer les informations reçues du casque sur une courte période dans une catégorie: concentré ou relaxé.

Une méthode *MetricValueGetter* a été ajoutée au code source pour y accéder.

La *metric value* évolue de façon continue et sans saccades tout en filtrant une partie du bruit et des artefacts⁸. Comparativement aux *times series*⁹, la *metric value* est moins sujette aux petites variations dans le signal, celles-ci n'étant pas forcément interprétées par le modèle comme des données représentatives d'une activité cognitive. Autrement dit, un travail de nettoyage du signal (bien qu'imparfait) est opéré par le modèle. Il se fait lors de l'attribution de la valeur de poids et de biais définissant "l'importance", estimée par le modèle, d'un feature¹⁰ donné dans le résultat de sa classification. Cette propriété de la *metric value* constitue un avantage pour le mapping (Figure 3).

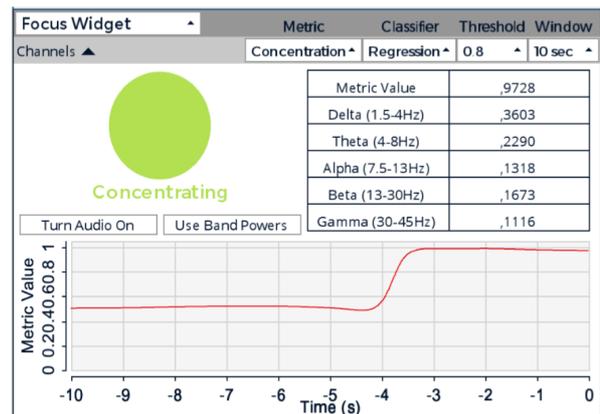


Figure 3. *Focus Widget* mesurant un état concentré depuis 3 secondes.

2.4. Définitions et utilisation de la bibliothèque *BrainFlow*

La version 5.1.0 d'*OpenBCI* fait désormais appel à la bibliothèque Java *BrainFlow*.¹¹ Cette bibliothèque est spécialisée dans l'analyse et le traitement de signaux biométriques et notamment des signaux EEG. C'est d'après les modèles d'apprentissage automatique de cette bibliothèque qu'est tirée la *metric value* et qu'est, par la même, établit la classification des états relaxé et concentré de l'utilisateur.

Au sein du *Focus Widget* d'*OpenBCI GUI* on peut ainsi faire appel à différents classifieurs¹², et accéder à certains paramètres comme la taille de la fenêtre d'analyse ou encore le seuil de confiance

⁸ Les artefacts sont du bruit résultant de phénomènes extérieurs à l'activité cognitive que l'on cherche à mesurer. (ie: clignement des yeux, serrage de mâchoires) ces derniers ont été une problématique non entièrement résolue du précédent projet.

⁹ Il s'agit de la tension électrique mesurée par chaque électrode en fonction du temps.

¹⁰ Données variables utilisées dans l'analyse du modèle.

¹¹ <https://brainflow.org/>

¹² Pour clarifier les interactions entre les différentes parties logiciels : la bibliothèque *BrainFlow* est utilisée ici par le logiciel *OpenBCI GUI*, au sein de celui-ci figure une fonction *Focus Widget* permettant de récupérer les états de concentrations au cours du temps.

nécessaire au classifieur pour considérer un état comme étant relaxé ou concentré.

D'après la documentation d'*OpenBCI*, l'état de relaxation est atteint en fermant les yeux et en faisant "de la méditation". L'état de concentration est atteint les yeux ouverts et en "se concentrant intensément".

Cette description étant un peu floue pour la concentration, nous avons mené nos propres tests pour essayer d'identifier les activités qui correspondent à l'état concentré discriminé par le *Focus Widget*.

Pour tous les tests menés, l'état de relaxation consistait à reproduire les consignes d'*OpenBCI* : fermer les yeux et faire des exercices de respiration.

Pour l'état de concentration, nous avons effectué différents exercices tels que du calcul mental, de la lecture de texte, de la lecture de notes, de la concentration visuelle¹³.

Chaque session de test a consisté en 5 minutes d'état relaxé puis cinq minutes réparties entre les différentes activités de concentration. Ces phases de tests seront discutées dans la conclusion de l'article.

Avant d'aller plus loin, et afin de clarifier nos choix de modèles, nous définirons ici brièvement l'apprentissage automatique.

L'apprentissage automatique, ou *machine learning* en anglais, est une branche de l'intelligence artificielle constituée d'un ensemble de techniques dont le but est d'apprendre à une machine à résoudre des problèmes ou à prendre des décisions en s'appuyant sur des données, sans être explicitement programmée pour cela [10]. En apprentissage automatique on cherche à définir des modèles mathématiques, des algorithmes, pouvant apprendre à partir d'un jeu de données dit d'entraînement, afin de réaliser des prédictions ou des classifications sur de nouvelles données.

Il existe différentes méthodes d'apprentissage pour entraîner un modèle : l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage par renforcement [11].

Le type d'apprentissage se choisit en fonction du type de données utilisées et des tâches à effectuer. Dans notre cas, classifier des états cognitif en "concentré" ou "relaxé", l'apprentissage supervisé est la seule méthode appropriée.

Plus spécifiquement, nous avons opté pour un modèle de régression logistique avec comme paramètres une fenêtre d'analyse de 10 secondes, et un seuil de 0,8 pour l'état concentré¹⁴ (0,2 pour l'état relaxé), les valeurs intermédiaires pouvant correspondre à un état "neutre".

Cette méthode statistique d'apprentissage supervisé est utilisée pour prédire si une certaine valeur est vraie ou fausse, en fonction d'autres variables connues, ici les valeurs des différents types d'ondes.

C'est à partir de cette méthode que sont classés les états mentaux en état concentré ou en état relaxé. Dans cette version du projet l'état mental est considéré comme binaire (selon la fonction (1)).

Cette régression utilise une fonction logistique¹⁵ pour modéliser la probabilité d'une réponse binaire – concentré ou relaxé – en fonction des variables explicatives. La régression logistique utilise également une fonction de coût, appelée fonction de perte logistique, pour minimiser l'erreur de prédiction. Cette fonction de coût¹⁶ prend en compte la distance entre les valeurs prédites et les valeurs réelles qui ont été établies lors de la phase labélisation des données par *BrainFlow*.

La régression logistique est donc un algorithme de classification binaire qui estime la probabilité qu'une observation appartienne à une catégorie en utilisant une fonction logistique et une fonction de coût pour ajuster les paramètres de la fonction de prédiction.

En d'autres termes, c'est à partir des données labellisées que l'algorithme va apprendre à différencier un état concentré d'un état relaxé, en apprenant à minimiser son erreur de classification. Une fois l'algorithme entraîné il est déployable sur des données non labellisées.

Entraîner un modèle d'apprentissage automatique signifie trouver le minimum de sa fonction de coût. Cela équivaut à réduire le nombre d'erreurs de classification moyen ce qui revient à améliorer la qualité de notre modèle dans la tâche de distinction des états mentaux concentré/relaxé.

L'entraînement d'un modèle s'établit à partir d'un jeu de données labellisées (qui sont les réponses qu'on cherche à obtenir de notre modèle) et des prédictions de l'algorithme qui après avoir été dans un premier temps complètement aléatoires s'affinent et se précisent simplement du fait de la minimisation de la fonction de coût. D'un point de vue technique, cette minimisation de la valeur de fonction de coût (proxy de l'erreur moyenne du modèle) se fait grâce à une descente de gradient [12]. Cette dernière informant le modèle de la participation et de l'importance de chacune des variables dans l'erreur produite par l'algorithme. Afin de bien entraîner un modèle il est important d'avoir un jeu de données d'entraînement représentatif des données réelles et nouvelles auxquelles l'algorithme aura affaire une fois déployé. Si ces données sont trop spécifiques ou trop peu spécifiques et que le modèle est mal paramétré on risque d'avoir un problème de sur-interprétation ou de sous-interprétation amenant à de mauvaises prédictions/classifications de l'algorithme. Dans notre cas, étant données que l'entraînement du modèle s'est fait à partir des données labellisées autres que celles issues des tests de notre utilisateur principal le risque de sur-interprétation est significatif. Ré-entraîner un modèle spécifiquement sur notre testeur

13 Exemple de test de concentration visuel réalisé : <https://www.youtube.com/watch?v=EI-b1fC4rVM>

14 Les valeurs de ces deux paramètres furent choisies empiriquement.

15 Egalement appelée fonction sigmoïde.

16 Le résultat de la fonction de coût donne une estimation de l'erreur du modèle.

principal est susceptible d'améliorer la fiabilité de nos classifications, mais également de rendre le modèle plus spécifique et le projet moins diffusable.

Comme précisé précédemment, nous avons opté pour l'algorithme de régression logistique au sein de ce projet. Cet algorithme se base sur une régression non linéaire et cherche à tracer une courbe sigmoïde pour coller au mieux aux données labellisées par *BrainFlow* de façon à maximiser le nombre d'occurrences dans lesquelles:

$$\begin{aligned} f(\text{paramètres des états concentrés}) &= 1 \text{ et} \\ f(\text{paramètres des états relaxés}) &= 0 \end{aligned} \quad (2)$$

Cette étape constitue la phase d'entraînement de notre modèle qui sera ensuite utilisé sur les données mesurées en temps réel afin de les associer à une classe.

Le résultat d'une fonction sigmoïde est toujours compris entre 0 et 1, et correspond à la *metric value*. Un résultat étant égal à 1 signifiant que l'état cognitif mesuré est identifié comme concentré et 0 comme relaxé. Toute valeur intermédiaire représente un certain niveau d'incertitude du classifieur (Figure 4).

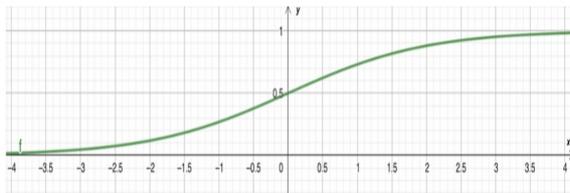


Figure 4. Courbe d'une fonction sigmoïde.

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3)$$

Tous les classifieurs disponibles sur *OpenBCI* sont des algorithmes d'apprentissage supervisé entraînés sur le même jeu de données labellisé, ayant lui-même été constitué par *BrainFlow*. Ce jeu de données n'est accessible que sur le site de *BrainFlow* et n'est pas en accès libre. Il en va de même pour les classifieurs de *BrainFlow* qui, même s'ils sont intégrés à un projet *Open Source*, ne sont pas libres d'accès. Ainsi, les différents classifieurs utilisés dans *OpenBCI* ne sont pas paramétrables, la plupart des paramètres sont inaccessibles même au niveau informationnel¹⁷. D'après la documentation *BrainFlow* il serait cependant possible d'ajouter des modèles ONNX¹⁸ au pipeline.

La phase de tests des différents classifieurs était d'autant plus cruciale dans le projet, du fait de la non

¹⁷ Par exemple, pour l'algorithme K-nearest neighbour (modèle disponible dans *brainflow*) on ne connaît pas la valeur de K et il est impossible de le modifier ce qui pourrait changer drastiquement les performances de ce classifieur.

accessibilité du jeu de données d'entraînement et du paramétrage des modèles.

Les classifications d'états de concentration et de relaxation ont été utilisées comme paramètres de la synthèse et de l'évolution du feedback sonore. L'idée étant de faire en sorte que l'utilisateur ait un retour sonore de son état mental plus intuitif que dans le projet auquel cet article fait suite.

Dans notre projet nous avons défini quatre états: relaxation profonde, relaxation, concentration, et concentration profonde sur lesquels nous reviendrons dans la partie "Aspects artistiques et sonores".

Le logiciel *Max/MSP* a été utilisé afin de réaliser la synthèse sonore, les données du système EEG sont ainsi communiquées à *Max/MSP* en OSC¹⁹ depuis le *OpenBCI GUI*. Pour des raisons d'optimisation, nous avons essayé de répartir le maximum de calculs sur le logiciel *OpenBCI GUI* pour que le patch *Max/MSP* ne se consacre qu'à la synthèse sonore et au mapping.

3. ASPECTS ARTISTIQUES ET SONORES

3.1. Synthèse soustractive et sonification

Pour ce qui est de la création sonore, nous utilisons le logiciel *Max/MSP*, dont la figure 4 présente le mode présentation du patch.

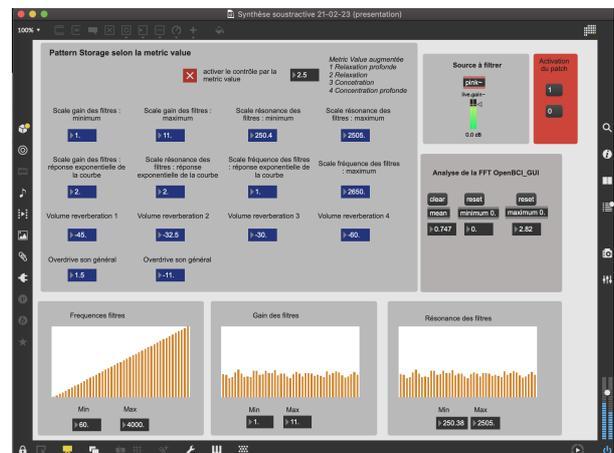


Figure 4. Mode présentation du patch *Max/MSP* utilisé pour la création sonore.

Nous nous basons sur un principe de synthèse soustractive. Cette méthode de synthèse possède l'avantage de produire un son riche plus facilement qu'une synthèse additive; elle procure aussi une approche originale de la sonification, tout en se

¹⁸ *Open Neural Network Exchange*: format d'export de modèle d'apprentissage automatique traduisible depuis la bibliothèque Python spécialisée *Keras*.

¹⁹ Protocole de communication *Open Sound Control*

combinant facilement avec les données d'une FFT. En effet, la FFT nous retournant des fréquences et des amplitudes, nous avons décidé de les associer aux paramètres de filtres en parallèle (objet : *fffb~*) appliqués à un bruit rose. C'est ensuite en modifiant les bornes de cette mise à l'échelle que le résultat sonore sera contrôlé via la *metric value*.

La FFT est calculée par le logiciel *OpenBCI GUI*, qui lui applique un filtre passe bande de 5 à 50 Hz. Elle est ensuite directement communiquée au patch *Max/MSP* avec un taux d'échantillonnage de 200 Hz. Ainsi nous n'utilisons que 45 valeurs de la FFT pour la synthèse sonore. Les fréquences de la FFT allant de 5 à 50 Hz, une partie non négligeable de celles-ci se trouve hors du champ audible (en dessous de 20 Hz) le reste étant extrêmement grave. Nous avons donc décidé de faire une mise à l'échelle linéaire avec 60 Hz comme fréquence minimum, la valeur maximum étant un paramètre variable contrôlé par la *metric value*.

L'amplitude du filtre et sa résonance sont mappées sur l'amplitude de chaque fréquence de la FFT. Ainsi lorsque l'amplitude est basse le son résultant est proche du bruit rose, le filtre laissant passer un grand nombre de fréquences, mais avec un volume faible. Lorsque l'amplitude est grande, le filtre devient plus résonant et son gain augmente, résultant en l'apparition d'une fréquence précise (Figure 5).

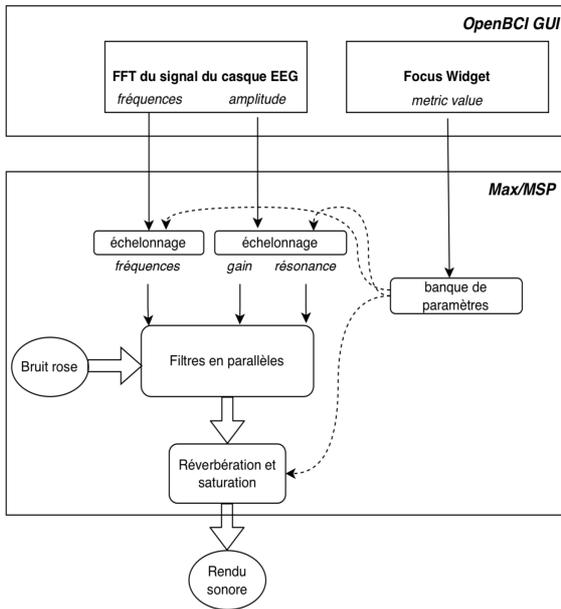


Figure 5. Schéma du fonctionnement du patch *Max/MSP*.

Le résultat d'ensemble s'entend comme un son homogène mais avec un contenu spectral toujours en variation.

Afin d'enrichir la création sonore et de créer des identités sonores distinctes pour chaque état mental, certains traitements sonores ont été ajoutés au résultat de la synthèse soustractive. Il s'agit d'une réverbération et

d'une saturation. La réverbération permet de polir le résultat assez brut de la synthèse sonore et est utilisée de façon intensive dans les états "profonds" pour mieux les caractériser. La saturation (objet : *overdrive~*) n'est présente que dans les états concentrés pour enrichir les sons issus des filtres résonants.

3.2. Contrôle de l'environnement sonore grâce à la *metric value*

Nous avons organisé le contrôle de l'environnement sonore selon quatre états mentaux (relaxation profonde, relaxation, concentration et concentration profonde).

La relaxation et la concentration profonde ne sont pas des états identifiés par le *Focus Widget*, il s'agit d'un ajout de notre part. L'accès à un état dit "profond" se fait en restant dans un même état avec un seuil de 0,2 – *metric value* en dessous de 0,2 pour l'état relaxé et au-dessus de 0,8 pour l'état concentré – pendant vingt secondes. Si l'état de l'auditeur change, le rendu sonore retourne à son précédent état en seulement trois secondes. Ainsi l'auditeur n'a accès aux états "profonds" qu'une fois qu'il maîtrise réellement son attention. L'intégration de ces états s'est faite afin de rendre le dispositif plus interactif mais aussi plus riche musicalement. Ces durées ont été choisies de façon empirique en faisant un compromis pour avoir un rendu musical intéressant sans trop contraindre le porteur du casque, rester vingt secondes d'affilée dans le même état n'étant pas une tâche trop ardue même sans entraînement.

Le contrôle du patch de création sonore par la *metric value* peut être résumé selon le schéma suivant. Les paramètres contrôlant ces états sont sauvegardés dans le patch (objet : *patrnstorage*). Pour passer d'un état à un autre, nous utilisons une interpolation linéaire qui suit la *metric value* entre l'état 2 et 3, et une interpolation linéaire en fonction du temps entre les états 1 et 2 et entre les états 3 et 4 (Figure 6).

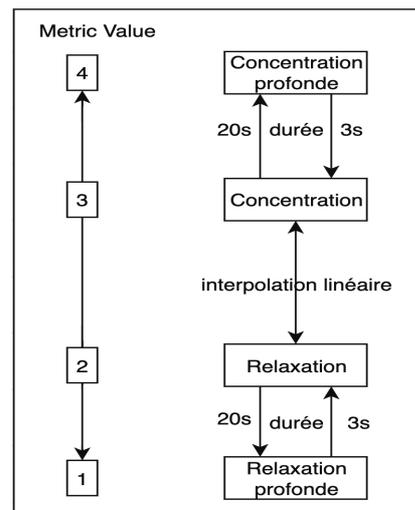


Figure 6. Schéma de la correspondance entre *metric value* et états de concentration à la base du contrôle de l'environnement sonore.

Au niveau sonore, le choix des paramètres ont été pensé pour donner les caractéristiques suivantes à chaque état de concentration. L'état de relaxation profonde correspond à de nombreux sons graves et doux dans une grande réverbération, qui peut faire penser au son de l'océan. L'état de relaxation est proche mais possède une plus grande gamme de fréquences. Quant aux états de concentration, ils sont caractérisés par des fréquences résonantes précises et aiguës, plongées dans une grande réverbération si l'état est "profond" (Tableau 1).

Paramètres	Relaxation profonde	Relaxation	Concentration	Concentration profonde
Gain minimum du filtre	0	1	1	0,01
Gain maximum du filtre	1	2	10	20
Résonance minimum du filtre	0.5	1	1000	3000
Résonance maximum du filtre	1	10	5000	5000
Fréquence maximum du filtre (Hz)	70	300	5000	7000
Overdrive	1	1	2	2,5
Réverbération	longue	courte	courte	longue

Table 1. L'ensemble des valeurs des paramètres qui caractérisent chaque état de l'utilisateur

L'idée englobant ce dispositif est de penser l'interaction, sous la forme d'une boucle de rétroaction, entre l'attention (symbolisée par le couple relaxation / concentration) du porteur du casque et l'environnement sonore, comme une forme musicale émergente. Pour ce faire, l'utilisateur doit pouvoir comprendre le fonctionnement du dispositif, ce qui implique un mapping en modulation temporelle avec lequel il peut se familiariser. Une fois cette étape réussie, il n'est pas exclu de créer une évolution temporelle du mapping – en accentuant les courbes de réponse des paramètres en fonction du temps par exemple – qui enrichira le rendu sonore tout en conservant une boucle de rétroaction claire.

4. DISCUSSION

Le projet est encore en phase de développement à plusieurs niveaux. Des tests portant sur d'autres classificateurs, sur le traitement des données et sur le rendu sonore sont encore en cours au moment de l'écriture de cet article.

Pour ce qui est du traitement des données, une étape de calibrage automatique est en cours d'élaboration. Cette étape semble nécessaire à chaque utilisation du dispositif puisque, selon la position des électrodes et le résultat des tests d'impédance, l'amplitude du signal du casque peut varier légèrement. Elle est aussi nécessaire afin de filtrer certains artefacts qui faussent les valeurs d'amplitude de la FFT, en faisant apparaître des amplitudes bien trop grandes pour la mise à l'échelle. Par exemple, des valeurs d'amplitude de plus de $300 \mu V$ pour des valeurs moyennes de $1 \mu V$.

Ce calibrage consisterait en la création d'une courbe de répartition des valeurs de la FFT sur une certaine durée. D'après les premiers tests effectués cette courbe est proche d'une gaussienne. Cependant, une étude statistique reste nécessaire afin de pouvoir fixer la borne supérieure des valeurs pour la mise à l'échelle. Avec ceci, les amplitudes de la FFT ne pourront plus dépasser cette valeur seuil, ce qui devrait améliorer de façon significative la mise à l'échelle.

Concernant la création de l'environnement sonore, plusieurs pistes d'approfondissements sont possibles. Le son de base utilisé dans la synthèse soustractive est ainsi sujet à évoluer, pour, par exemple, passer d'un bruit rose à un son issu de synthèse granulaire.

De plus, pour renforcer l'identité sonore de l'état de concentration profonde ainsi que sa compréhension par l'auditeur, l'idée d'ajouter un/des opérateur(s) comme un LFO (*Low Frequency Oscillator*) sur le rendu sonore est en voie d'implémentation. Un LFO pourrait par exemple s'activer uniquement dans les dernières secondes avant l'état "profond" afin d'indiquer au porteur du casque de façon claire son changement d'état tout en amenant de la variabilité aux rendus sonores possibles du dispositif.

Dans le même ordre d'idée, le paramétrage de la synthèse soustractive nécessite encore certains tests pour affiner l'esthétique du rendu sonore et son aspect intuitif à retranscrire un état relaxé ou concentré.

Pour ce qui est du contrôle de l'environnement sonore par la *metric value*, une des pistes à explorer est l'intégration, pour certains paramètres, d'interpolations exponentielles ou bien de courbes de réponse dessinées sur mesure afin de sculpter des transitions sonores plus subtiles, chaque paramètre ayant sa propre évolution selon la *metric value*.

Par ailleurs, la recherche d'ERP (*event-related potential*) en lien avec l'environnement sonore reste une partie annexe de ce projet. Les ERP ou potentiels évoqués, sont ainsi des réponses corticales fortes mesurables à la suite d'un stimuli précis. La mise en place de tels stimuli (ici sonore) n'est pour l'instant pas faite dans le projet, mais une recherche en ce sens est en cours.

Enfin, afin d'avoir un meilleur contrôle paramétrique sur les modèles utilisés dans la classification de données, il est envisagé d'importer nos propres

classifieurs au format ONNX à travers *BrainFlow*. En effet, bien que la classification des états concentré/relaxé fonctionne empiriquement, tous les classifieurs disponibles nativement dans *BrainFlow* utilisent des méthodes d'apprentissage supervisé entraînés sur un jeu de données n'étant pas accessible librement. Pour l'heure, il semble impossible de faire un "fine tuning"²⁰ des modèles. Or, même si le système semble marcher en l'état, il existe des variations probables des activités corticales selon les utilisateurs, justifiant une étape de fine tuning. Cette étape pourrait potentiellement amener de sensibles améliorations dans les classifications des modèles. L'utilisation de *BrainFlow* par OpenBCI nous amène à nous poser des questions sur la méthodologie mise en œuvre pour l'entraînement des classifieurs, sur la relative opacité des paramètres de certains modèles ainsi que sur la constitution du jeu d'entraînement et de ses données labellisées.

5. CONCLUSION

Cette seconde version du projet est encourageante vis-à-vis des objectifs fixés. L'environnement sonore réagit de façon convenable aux variations de la *metric value* et celle-ci semble bien corrélée avec l'état mental du porteur du casque selon les tests effectués. De plus, l'environnement sonore créé est suffisamment intuitif pour créer une boucle de rétroaction avec l'utilisateur, transformant le casque en objet plus complexe qu'un simple contrôleur. Bien qu'encore imparfait, le feedback du dispositif est ainsi plus intuitif que dans la première version grâce à la retranscription sonore du niveau de concentration/relaxation. Un autre apport de ce projet comparativement au précédent réside dans la sensible réduction du bruit et des artefacts issues des mesures du casque. Ceci a été rendu possible en utilisant une métrique d'ordre supérieure aux time series, la *metric value* des classifieurs [13].

Finalement, sur le plan théorique, ce projet pose des questions à la fois sur la sonification et l'attention : jusqu'où peut-on associer attention et couple d'état relaxé/concentré ? Quels sont les différents types d'interaction entre le porteur du casque et l'environnement sonore ? Et comment les enrichir ?

Autant de questions auxquelles le projet tentera de répondre lors de son développement.

6. RÉFÉRENCES

[1] Meric, R. et Solomos, M. "Audible Ecosystems, and emergent sound structures in Di Scipio's music. Music philosophy enhances musical analysis", Fourth Conference for Interdisciplinary Musicology (CIM08), Thessaloniki, Greece, 2008.

- [2] Schoon, Andi and Dombois, F. "Sonification in music." in *Proceedings of the 15th International Conference on Auditory Display*, Copenhagen, Denmark, 2009.
- [3] Giomi, A. « Du field recording à l'art de la sonification ». *Filigrane*, Paris, 2022.
- [4] Polli, A. « Atmospherics/weather works: A spatialized meteorological data sonification project ». *Leonardo*, 2005, vol. 38, no 1, p. 31-36, .
- [5] Abenavoli, L. « The Pulse of the Earth and sonification » . *AI & society*, 2012, vol.27, p.277-279.
- [6] Citton, Y. *Pour une écologie de l'attention*, Paris, Seuil, 2014.
- [7] Bateson, G. *Vers une écologie de l'esprit* (Drosso, F., Lot, L., Simion, E., Trad.), Paris, Seuil, 1977.
- [8] Bazin, A., et Del Ghingaro, U. « Projet CAANEVA, Possibilités et limitations de L'usage Des Interfaces Cerveau-Machine Dans un Contexte de Création d'environnement visuel et sonore ». *Acte musical et environnements informatiques : actes des Journées d'Informatique Musicale 2020*, Université de Strasbourg/Faculté des Arts, Strasbourg, 2021
- [9] Jasper, H. "Report of the committee on methods of clinical examination in electroencephalography", *Electroencephalography and Clinical Neurophysiology*, USA, 1958.
- [10] Bishop, C. M. *Pattern recognition and machine learning*. Springer, Berlin, Allemagne, 2006
- [11] Ben-David, S. & Shaleve-Shwartz, S. *Understanding machine learning from theory to algorithms*. Cambridge University Press, Cambridge, Angleterre, 2014.
- [12] Bengio, Y. ; Courville, A. & Goodfellow, I. *Deep Learning*. The MIT Press, Cambridge, USA, 2016
- [13] Nunez, P. L., & Srinivasan, R. *Electric fields of the brain: the neurophysics of EEG*. Oxford University Press, USA, 2006.

²⁰ Ajustement de modèles pré-entraînés à l'aide de données issues de mesures personnelles.