

EEGSea

Configuration et communication entre les blocs du projet

Soufyane Janati-Idrissi
Jonathan Pennec
Mathieu Quentel
Adrien Vernotte

Sommaire

A. Interface du casque en Python.....	3
A.1) Configuration et exécution.....	3
A.2) Format des trames envoyées au projet Unity.....	3
B. Interface graphique de configuration des vagues.....	3
B.1) Différents environnements Unity.....	3
B.2) Utilisation de l'interface.....	4
a) Sélection d'un groupe d'électrodes prédéfini.....	4
b) Sélection d'une plage de fréquence prédéfinie.....	4
c) Saisie et visualisation des fréquences.....	4
d) Saisie et visualisation des électrodes.....	5
e) Visualisation des groupes.....	5
B.3) Format des données de configuration.....	5
B.4) Configuration des vagues.....	6

A. Interface du casque en Python

Les données sont reçues du casque EEG par le programme Python `AquisitionEEGclientUDPandOSC.py`, qui les transmet au projet Unity par la boucle locale (UDP, port 5000). Les données sont groupées dans une file, puis envoyées en une seule fois à rythme régulier.

A.1) Configuration et exécution

Python 2.7 est nécessaire pour exécuter ce script. L'installation des dépendances requises et les commandes pour exécuter le projet sont détaillées dans `README.md` et `README_Linux.md`.

Le script peut être modifié pour régler les données suivantes :

- Fréquence d'envoi des trames UDP (Hz) : `udp_freq = 128`

Cette fréquence limite la fréquence maximale mesurable par la suite (Théorème de Shannon : `max_freq = udp_freq/2`).

- Taille du buffer de données : `buf_length = 10`

Si un retard s'accumule dans la réception des données, augmenter cette valeur peut résoudre le problème. En contrepartie, un retard statique sera ajouté à chaque augmentation de la taille du buffer : `retard=buf_length/udp_freq`. Pour un buffer de dix unités de temps (valeur par défaut), le retard statique théorique est de 78 ms.

A.2) Format des trames envoyées au projet Unity

Pour chaque unité de temps, une liste contenant la valeur du temps en seconde et le niveau des capteurs est créée. Les listes sont ensuite groupées pour être envoyées ensemble, au format suivant :

```
temps;nomCapteur:valeur;nomCapteur:valeur;nomCapteur:valeur;[...] |
temps;nomCapteur:valeur;nomCapteur:valeur;nomCapteur:valeur|[...]
```

- Les unités temporelles sont séparées par des `|` dans le buffer.
- Les données de capteur sont séparées par des `;` dans l'unité temporelle, le premier élément est le temps en secondes.
- Le nom du capteur et sa valeur sont séparés par un `:` dans la donnée de capteur.

B. Interface graphique de configuration des vagues

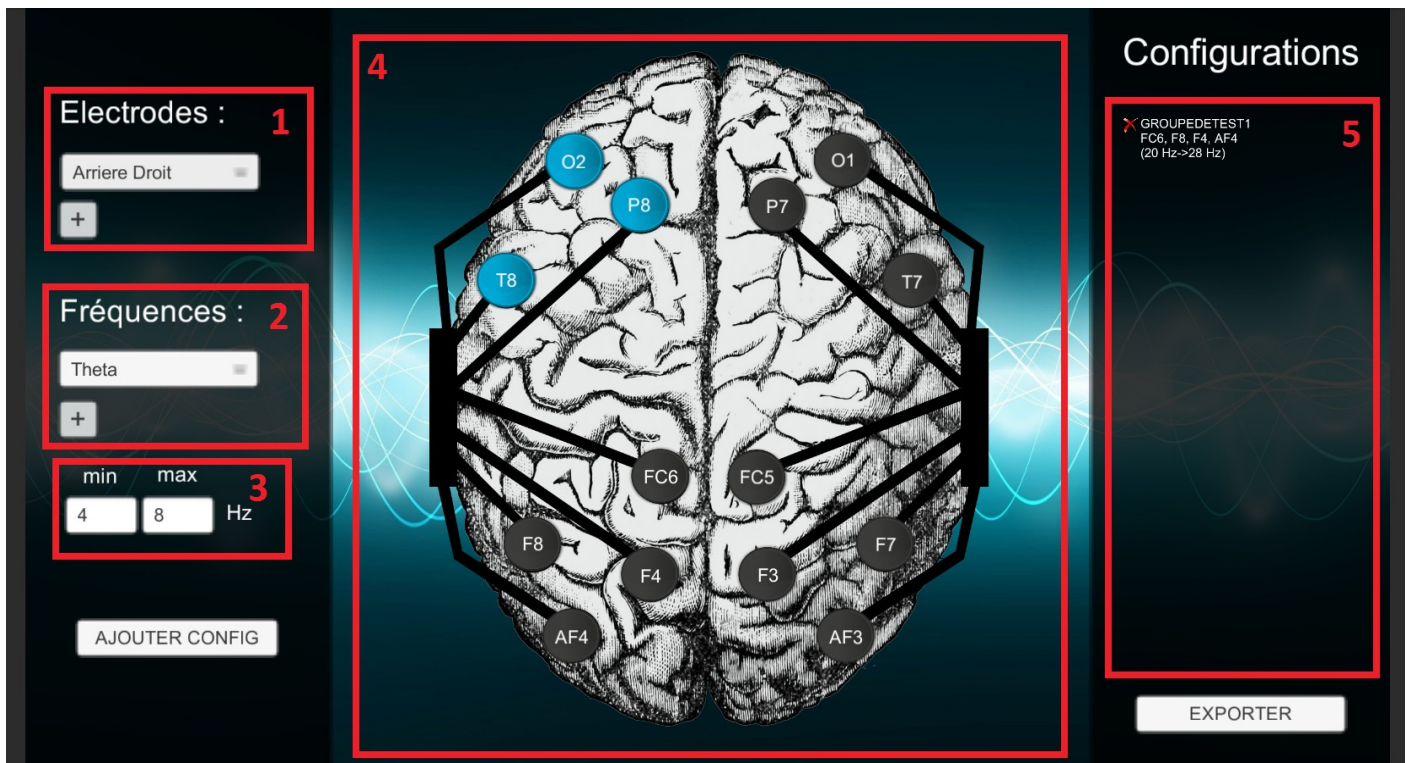
B.1) Différents environnements Unity

L'interface graphique de configuration des vagues n'est disponible que pour les versions du projet générées avec Unity 5.2. Typiquement, ce n'est pas le cas sous Linux actuellement.

Cette interface génère un fichier de configuration au format XML, qui peut être édité directement sur les plate-formes ne supportant pas Unity 5.2.

B.2) Utilisation de l'interface

Une interface a été développée afin de créer des groupes d'électrodes et de les associer à des plages de fréquences.



a) Sélection d'un groupe d'électrodes prédéfini

La liste déroulante permet de sélectionner des groupes prédéfinis. Ces groupes sont stockés dans le fichier de configuration "merUnity\Assets\MerMiroir\Config\groups.txt". Lorsqu'un groupe prédéfini est sélectionné, les électrodes du groupe sont colorées en bleu (zone 4).

Pour ajouter un groupe d'électrodes, il faut sélectionner les électrodes sur le crâne, et enregistrer le groupe en utilisant le bouton (+). Il faut ensuite saisir un nom pour ce nouveau groupe.

b) Sélection d'une plage de fréquence prédéfinie

La liste déroulante permet de sélectionner des plages de fréquences prédéfinies. Ces plages sont stockées dans le fichier de configuration "merUnity\Assets\MerMiroir\Config\freqs.txt". Lorsqu'une plage prédéfinie est sélectionnée, les fréquences min et max sont modifiées (zone 3).

Pour ajouter une plage de fréquences, il faut rentrer les valeurs des fréquences min et max, et enregistrer la plage en utilisant le bouton (+). Il faut ensuite saisir un nom pour cette nouvelle plage.

c) Saisie et visualisation des fréquences

Permet de sélectionner une plage de fréquence ou de visualiser la plage prédéfinie sélectionnée.

d) Saisie et visualisation des électrodes

Permet de sélectionner des électrodes ou de visualiser le groupe prédéfini sélectionné.

e) Visualisation des groupes

Permet de visualiser les groupes qui ont été ajoutés (bouton « AJOUTER CONFIG »). Il est alors possible de supprimer des groupes ou d'exporter la configuration au format XML (bouton « EXPORTER »)

B.3) Format des données de configuration

Les fichiers de configuration sont des fichiers XML. Ils contiennent les groupes définis avec l'interface. Les fichiers XML sont normalisés par des fichiers de XML schema (fichiers XSD), cela permet de valider leur structure et leur contenu.

Le fichier XSD suivant définit les normes des fichiers acceptés par le programme.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <!-- CONFIG -->
  <xsd:element name="config">
    <xsd:complexType>
      <xsd:sequence>
        <!-- GROUP -->
        <xsd:element name="group" minOccurs="1"
          maxOccurs="unbounded">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="name"
                type="xsd:string"/>
              <xsd:element name="f_min"
                type="xsd:int"/>
              <xsd:element name="f_max"
                type="xsd:int"/>
            <!-- LIST OF ELECTRODES -->
            <xsd:element name="electrodes_list">
              <xsd:complexType>
                <xsd:sequence>
                  <!-- ELECTRODE -->
                  <xsd:element
                    name="electrode"
                    type="xsd:string"
                    minOccurs="1"
                    maxOccurs="14"/>
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:schema>
```

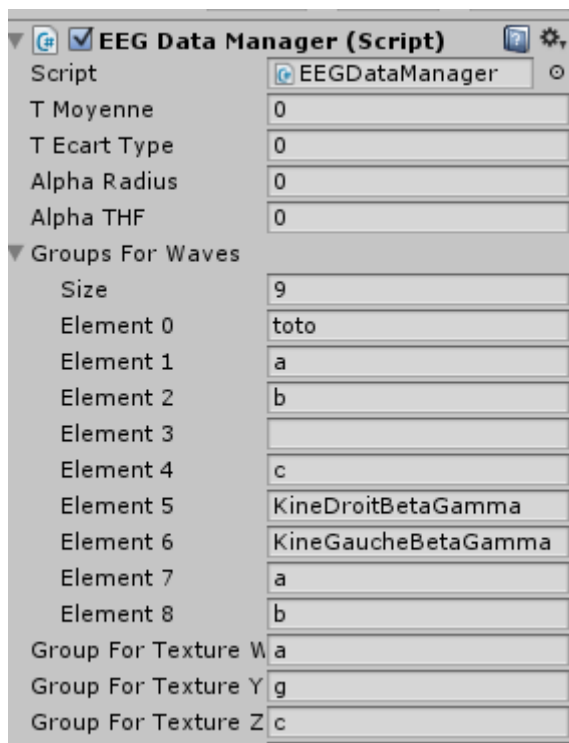
En résumé, les fichiers de configuration doivent être composés d'une balise principale `<config>` qui peut contenir plusieurs `<group>`.

Un `<group>` doit contenir les balises `<name>`, `<f_min>`, `<f_max>` et `<electrodes_list>`.

La balise `<electrodes_list>` peut contenir de 1 à 14 balises `<electrodes>`.

B.4) Configuration des vagues

Les vagues peuvent être configurées pour correspondre à une configuration dans Unity Editor. L'objet à configurer est l'EEGDataManager. Pour chaque vague, ainsi que pour les propriétés texture (W,Y,Z) il faut saisir le nom du groupe préalablement défini dans les fichiers XML.



Groupes pour les vagues 0 à 7

Groupes pour les propriétés textures