

EEGDiag, une application d'analyse de l'EEG pour la plateforme de télésanté BBEEG.

Pierre Chauvet¹, Sylvie Nguyen²

¹LISA EA 4094, LUNAM Université, Université Catholique de l'Ouest, 3 place André-Leroy BP 10808, 49008 Angers - France

²LISA EA 4094, LUNAM Université, Centre Hospitalier Universitaire, 4 rue Larrey, 49000 Angers - France

pierre.chauvet@uco.fr , sylvie.nguyenthe@gmail.com

Resumé

L'application Java EEGDiag permet de visualiser et analyser des électroencéphalogrammes (EEG), en complément des autres modules de la plateforme BBEEG. Cette plateforme est un portail internet pour la télé-expertise, le télédiagnostic et le e-learning dans le domaine de l'interprétation de l'EEG du nouveau-né, qui présente une valeur diagnostique et pronostique reconnue d'une évolution anormale du cerveau. EEGDiag est basée sur une approche par composants des algorithmes d'analyse du signal, c'est-à-dire comme une séquence de modules prédéfinis. L'application est évolutive car basée sur des mécanismes de type plug-in. Enfin, nous proposons 4 algorithmes de détection des intervalles interbouffées (IBI) et effectuons leur analyse en se basant sur le classement de 408 EEG avec des critères d'inertie intra-classe et de distance inter-classe.

Mots-clés: *Electroencéphalogramme, Analyse du signal, Classification, Intervalle interbouffée.*

1 Introduction

La plateforme BBEEG est un portail internet offrant différents services autour de l'interprétation de l'électroencéphalogramme (EEG) du nouveau-né. Cet outil de télésanté, développé par un consortium public-privé dans le cadre de la réponse à l'appel d'offre de l'Agence Nationale de la Recherche TECSAN 2010, intègre les fonctionnalités permettant le télédiagnostic et la télé-expertise, le e-learning et l'analyse automatisée de l'EEG selon certains marqueurs. Il s'agit de mettre à disposition des praticiens via un portail unique:

- l'accès à un diagnostic médical ou à un avis d'expert en EEG du nouveau-né, de façon sécurisée via un service de télé-médecine,
- des tracés EEG de référence normaux et anormaux, ainsi qu'à des cours sur le sujet,
- des outils d'aide au diagnostic par analyse du signal.

L'EEG est une technologie d'imagerie non invasive et peu coûteuse de la fonction électrophysiologique cérébrale, qui mesure l'activité électrique du cerveau correspondant au courant généré dans les neurones. Cette activité est enregistrée à l'aide de capteurs posés sur le cuir chevelu. Chez le nouveau-né prématuré l'EEG a la particularité d'être physiologiquement discontinu, constitué d'une alternance de périodes d'activité (les Bursts) et de périodes d'aplatissement appelées intervalles inter-bouffées (IBI ou Interburst Interval – voir figure 1), aspect pathologique

à tout autre moment de la vie. Cet aspect évolue très rapidement au cours de la période de prématurité (24 à 37 semaines) et de la même façon in utero et ex utero. L'EEG permet donc de donner un âge maturatif à l'enfant, âge qui peut ou pas concorder avec le terme théorique. La valeur pronostique de l'EEG a été démontrée dans l'étude EPIPAGE malgré des critères d'analyse très généraux de l'EEG [1], et la valeur diagnostique et pronostique des anomalies de l'EEG néonatal chez l'enfant prématuré est bien établie [2]. En particulier, il a été montré que la durée du plus long IBI dans l'EEG est fortement corrélée avec une évolution anormale du cerveau [3], et que la durée des IBI est modifiée par l'administration de morphine chez le nouveau-né prématuré [4].

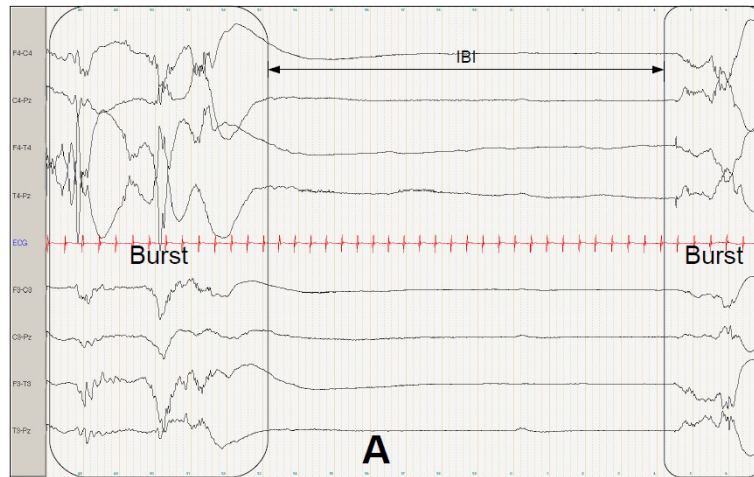


Figure 1: Une page d'EEG, avec deux phases d'activités (Bursts) séparées par un intervalle inter-bouffées (IBI - partie A)

Ainsi, l'IBI est un marqueur très prometteur pour le calcul d'indices de risque sur le devenir de l'enfant. Malheureusement la diffusion de cette approche est freinée par l'hyperspécialisation de la technique de l'EEG néonatal, la pénurie d'experts du domaine qui ont des difficultés à assurer la transmission des connaissances, et la durée de l'analyse visuelle elle-même sur des enregistrements de plus en plus long (30mn à 48h). La plateforme BBEEG est construite pour répondre à ces problèmes, en alliant la pratique courante (le portail de télédiagnostic) à la transmission du savoir (le module d'E-learning) et à la recherche pour l'analyse automatique de l'EEG et le calcul d'indices de risque. Le développement d'outils d'analyse du signal est donc intégré au projet global d'aide au diagnostic en EEG du nouveau-né. Cela implique des contraintes fortes : rapidité d'exécution, simplicité d'utilisation pour des praticiens utilisateurs occasionnels de la plate-forme, capacité à s'enrichir de nouvelles méthodes.

Par conséquent nous avons pris la décision de développer conjointement avec le portail une application libre Java, appelée EEGDiag, installée et exécutée sur le poste client à partir du navigateur. La version actuelle permet à la fois pour le chercheur de tester des algorithmes décrits sous forme de pipe-line d'analyse et pour le praticien de l'aider à construire un diagnostic, voire un pronostic, à partir des variables extraites de l'EEG.

2 L'application EEGDiag

Nous avons vu en introduction que le nombre et la durée des intervalles inter-bouffées (IBI) dans l'EEG est un marqueur important de la maturation cérébrale. Ils peuvent être repérés et décomptés

plus ou moins correctement selon les algorithmes utilisés. Il est alors possible, à partir des mesures obtenues par binarisation des signaux EEG en phases d'activité et en IBI (telles que le nombre d'IBI, la durée moyenne des IBI, la durée de l'IBI le plus long...) de fournir des indices d'un risque de développement cérébral anormal. Nous avons donc développé l'application EEGDiag en Java (pour faciliter l'intégration dans le portail) dont les objectifs sont :

- d'implémenter des algorithmes de détection des IBI, de les tester puis de les comparer sur différents EEG,
- d'établir pour chaque algorithme un paramétrage par défaut fonctionnant dans le plus grand nombre de cas,
- de construire des indices d'analyse à partir des IBI détectés (pourcentage d'IBI, complexité de Lempel-Ziv, entropie, classification par réseau neuronal...) pertinents pour le neuropédiatre,
- de restituer les résultats (textes et graphiques) de ces analyses.

Un objectif non moins important pour rendre l'application évolutive, attrayante pour le chercheur et l'apprenant, est de décomposer ces algorithmes en un certain nombre d'opérations élémentaires de manière à pouvoir les modéliser par un graphe décrivant l'enchaînement de ces opérations. Ceci nous a permis de réaliser un outil de modélisation graphique des algorithmes pour des utilisateurs non informaticiens. De plus, cette approche par composants (un composant, ou module, est chargé d'une opération élémentaire) permet une meilleure réutilisabilité de nos algorithmes, facilite l'intégration de nouveaux modules et le travail de comparaison entre algorithmes.

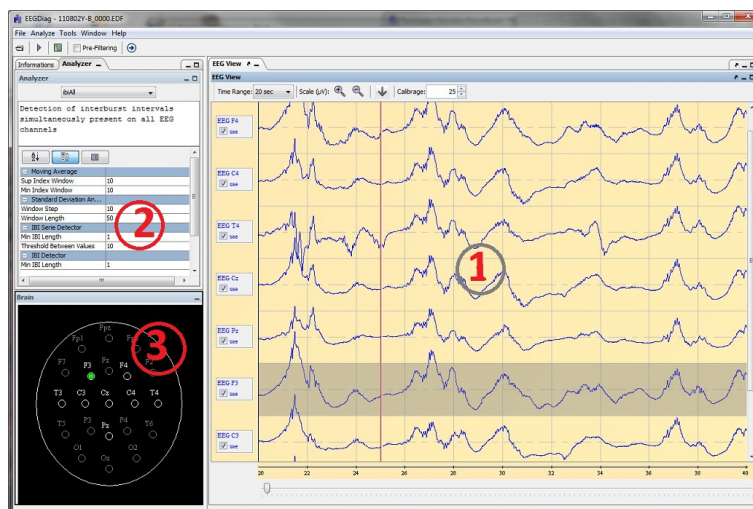


Figure 2: Les principaux panneaux de l'application EEGDiag : (1) affichage de l'EEG à analyser avec coloration des plages IBI si elles sont détectées ; (2) choix de l'analyseur et, pour l'analyseur choisi, boîte de propriétés ; (3) représentation des électrodes sur le scalp, électrodes actives en surbrillance

L'application EEGDiag, lancée à partir du portail BBEEG, permet d'afficher des EEG en différents formats, d'appliquer sur ces EEG des analyseurs prédéfinis ou construits par l'utilisateur, et de visualiser les résultats de l'analyse (voir figure 2). Elle peut être utilisée en routine, ou pour essayer de nouveaux analyseurs. Enfin, une gestion dynamique des modules d'analyse et un menu "outils" permet de l'enrichir de nouvelles fonctions en se basant sur un mécanisme de plug-in : à

condition de respecter un framework bien défini, l'utilisateur peut ajouter de nouveaux modules et de nouveaux outils sous forme de bibliothèque Java (format *.jar). L'un des outils livré de base permet d'appliquer différents analyseurs de détection d'IBI sur un lot de fichier EEG, et d'exporter les résultats du lot d'analyse dans un fichier csv.

EEGDiag est composée d'un certain nombre de librairies Java construites dans le cadre du projet (voir figure 3 pour leur graphe d'interactions). SeriesCommon.jar est la bibliothèque de base, puisqu'elle contient l'ensemble des structures de données permettant la manipulation de séries temporelles et de séries d'intervalles (lecture/écriture entre une zone tampon avec double buffer et les fichiers, les EEG étant trop longs pour être chargés intégralement en mémoire) ainsi que l'affichage de ces signaux, en particulier le visualiseur EEG. BBEEG-AnalyzerFramework.jar et BBEEG-ToolFramework.jar sont les frameworks permettant d'enrichir l'application de nouveaux modules (bibliothèques jar), de nouveaux algorithmes (fichiers XML) et de nouveaux outils (bibliothèques jar) et bien sûr de les utiliser. Plusieurs librairies libres disponibles sur Internet sont employées, en particulier JSPF (Java Simple Plugin Framework, voir [5]) pour avoir un chargement dynamique des modules et des outils de type plug-in, et BSAF (Better Swing Application Framework, voir [6]) pour simplifier la séparation entre les aspects fonctionnels et l'interface.

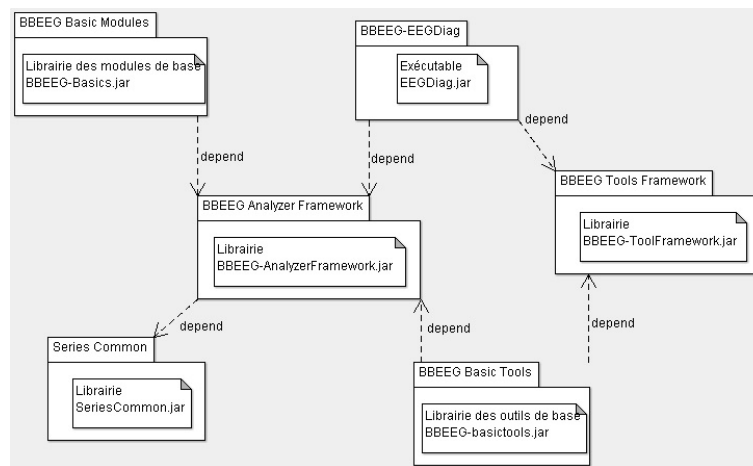


Figure 3: Dépendances entre bibliothèques java - BBEEG-EEGDiag est le projet permettant de générer l'application EEGDiag, qui dépend directement des bibliothèques BBEEG-AnalyzerFramework.jar et BBEEG-ToolFramework.jar; les bibliothèques BBEEG-Basics.jar (regroupant les modules de base) et BBEEG-basictools.jar (regroupant quelques outils) ne sont chargées qu'au lancement de EEGDiag grâce à la librairie de gestion de plug-in JSPF

3 La pipeline d'analyse de l'EEG

L'analyse du signal est rarement basée sur un algorithme simple et unique. Il peut être décrit par un ensemble d'opérations séquentielles permettant l'élimination d'informations inutiles et du bruit, de l'extraction d'informations provenant de différentes sources, de leur agrégation et de leur transformation sous une forme concise, facile à interpréter pour le praticien. Nous avons implémenté cette pipeline d'analyse, que nous appelons dans la suite analyseur, sous forme d'un vecteur ordonné de modules parcouru de manière séquentielle. Chaque module correspond à une opération sur l'EEG ou des données dérivées : une transformation homothétique du signal (filtrage, lissage), l'analyse du signal elle-même (détection des IBI, décomposition dans le domaine fréquentiel), l'extraction de certaines caractéristiques (complexité, entropie), et finalement de la classification

pour l'analyse de risque. Ainsi un analyseur réalise une analyse séquentielle de l'EEG où chaque module termine son opération avant d'envoyer son résultat au module suivant. Un analyseur peut être décrit graphiquement par un organigramme (correspondant à un schéma de procédé, voir figure 4) où :

- chaque boîte représente un module sélectionné parmi la liste des modules disponibles (automatiquement mise à jour au lancement de EEGDiag par un mécanisme de plug-in), avec l'IHM permettant de définir son paramétrage par défaut,
- chaque arc orienté représente l'ordre de communiquer l'objet résultat du module source (sa sortie) à l'entrée du module puits, ce qui indique implicitement l'ordre d'exécution des modules (ordre topologique sur le graphe)

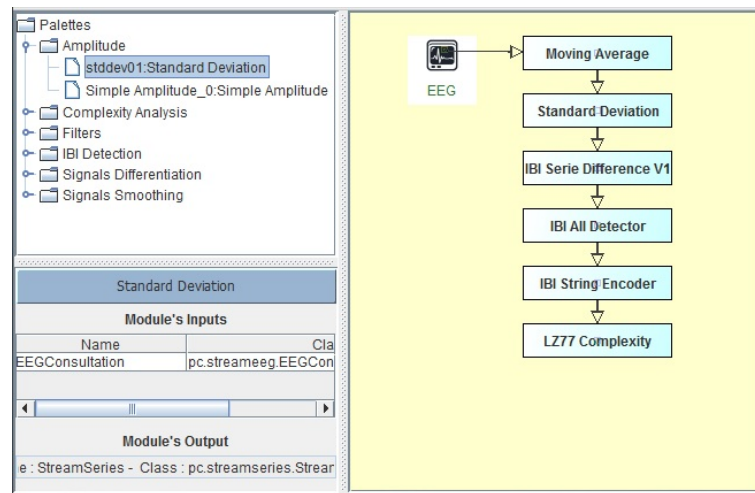


Figure 4: Exemple d'analyseur (calcul de la complexité de Lempel-Ziv de l'EEG binarisé par détection des IBI) représenté par son organigramme dans l'outil de modélisation graphique de EEGDiag ; panneau en haut à gauche : liste des modules disponibles (classés dans un arbre); panneau en bas à gauche : détail du module sélectionné; panneau à droite : surface de construction de l'organigramme

L'algorithme appliqué par un analyseur consiste simplement à exécuter l'un après l'autre les modules rangés dans une liste selon leur ordre topologique :

- Initialisation du rapport (texte).
- $Current$ = le premier module dans la liste.
- $Input(Current)$ = la source de données définie par le lecteur d'EEG.
- Exécuter le module $Current$ et ajouter au rapport le rapport produit par $Current$.
- Tant que l'analyseur est dans l'état "Exécution" ET que $Current$ n'est pas le dernier module dans la liste Faire:
 - $Last = Current$;
 - $Current$ = prochain module dans la liste;
 - $Input(Current) = Output>Last$;

- Exécuter *Current* et ajouter au rapport le rapport produit par *Current*;
- Placer l'analyseur en état "Stoppe" et retourner le rapport.

Toutes les données définies lors de la construction de cet organigramme sont sauvegardées dans un fichier texte au format XML, qui fournit la liste des modules à charger et l'ordre de leur exécution. Ce fichier XML (voir figure 5) peut d'ailleurs être construit sans l'outil graphique, même si cela peut être source d'erreurs.

```
<?xml version="1.0" standalone="true"?>
<!DOCTYPE ANALYZER_MODEL SYSTEM "analyzermodel.DTD">
- <EEG_ANALYZER_MODEL type="sequential" name="IbiAll_Simple">
  <COMMENTS>Detection of interburst intervals simultaneously present on all
  EEG channels using a threshold on amplitude to build an IBI</COMMENTS>
  <HTML>No_Comment_IM.htm</HTML>
  <TAG>IBIDETECTOR</TAG>
  <DATASOURCES>
    <SOURCE name="eeg" class="pc.streameeg.EEGConsultation"/>
  </DATASOURCES>
  <MODULES>
    - <MODULE name="smooth01"
      class="bbeeg.modules.smoothing.MovingAverage">
      <SHOWOUTPUT ownview="no"/>
      <PARAMETER name="infIndex" published="yes" value="10"/>
      <PARAMETER name="supIndex" published="yes" value="10"/>
    </MODULE>
    - <MODULE name="stddev01"
      class="bbeeg.modules.amplitude.StdDeviation">
      <SHOWOUTPUT ownview="no"/>
      <PARAMETER name="windowStep" published="yes" value="10"/>
      <PARAMETER name="windowLength" published="yes" value="50"/>
    </MODULE>
    - <MODULE name="ibidetector01" class="bbeeg.modules.ibi.IBISerieSimple">
      <PARAMETER name="threshold" published="yes" value="20.0f"/>
      <PARAMETER name="minIbiLength" published="yes" value="1.0f"/>
    </MODULE>
    - <MODULE name="IbiAll01" class="bbeeg.modules.ibi.IBIAllDetector">
      <SHOWOUTPUT ownview="yes"/>
      <PARAMETER name="minIbiLength" published="yes" value="1.0f"/>
    </MODULE>
  </MODULES>
  <MAINVIEW action="complete" module="IbiAll01"/>
</EEG_ANALYZER_MODEL>
```

Figure 5: Exemple de fichier XML décrivant un analyseur, ici pour la détection des IBI

4 Les algorithmes de détection des IBI

L'idée sur laquelle nous nous sommes basés pour construire nos algorithmes de détection automatique des IBI est d'imiter l'analyse visuelle réalisée par le médecin. Les discussions avec les neurophysiologistes ont montré que la classification visuelle d'un IBI est basée essentiellement sur l'aspect plat de la ligne de base de l'EEG entre deux bursts, plus que par une mesure systématique de l'amplitude des pics. En conséquence, nos analyseurs ont été définis pour étudier sur chaque canal de l'EEG la variation de l'estimation de l'écart-type entre de courtes fenêtres contigües et se recouvrant partiellement. Un IBI est détecté si cette variation est inférieure à un seuil donné en microVolts et si la fenêtre temporelle est suffisamment longue. Les IBI sur l'ensemble de l'EEG sont les intersections des IBI détectés sur chaque canal. Plusieurs modules doivent être exécutés pour détecter les IBI sur l'ensemble de l'EEG, et l'ajout de certains modules peut améliorer la qualité du résultat.

Parmi l'ensemble des modules implémentés, voici ceux utilisés le plus fréquemment pour détecter les IBI:

1. Module *Filter*: chaque canal est filtré à 50Hz (fréquence de distribution de l'électricité en Union Européenne) avec un filtre IIR de Butterworth du second ordre.
2. Module *Smoother*: chaque canal est lissé avec une fenêtre mobile par la moyenne simple, une moyenne pondérée ou la médiane. Pour la détection des IBI la moyenne mobile simple s'est avérée suffisante.

3. Module *Standard Deviation*: chaque canal est parcouru en calculant une estimation de l'écart-type sur des fenêtres successives partiellement recouvrantes, pour produire des séries d'écart-types.
4. Module *IBI Channel Detector / Standard deviation Threshold*: algorithme constitué de 3 phases; (i) pour chaque série d'écart-types, si l'écart-type est inférieur à un certain seuil V_T l'intervalle de temps correspondant est agrégé avec l'intervalle de l'écart-type précédent (s'il existe, sinon un nouvel IBI est créé); (ii) regrouper tous les intervalles séparés de moins de m2 secondes; (iii) éliminer tous les intervalles de moins de m1 secondes.
5. Module *IBI Channel Detector / Standard deviation Variation Threshold*: algorithme constitué de 2 phases; (i) pour chaque série d'écart-types, si la différence entre 2 valeurs successives est inférieure à un certain seuil V_T les intervalles de temps correspondants sont agrégés, et un IBI est créé s'il dure au moins m1 secondes; (iii) les IBI séparés par moins de m2 secondes sont agrégés.
6. Module *IBI EEG Detector*: calcul des intersections entre chaque série d'IBI, puis élimination de tous les IBI qui durent moins de m3 secondes.

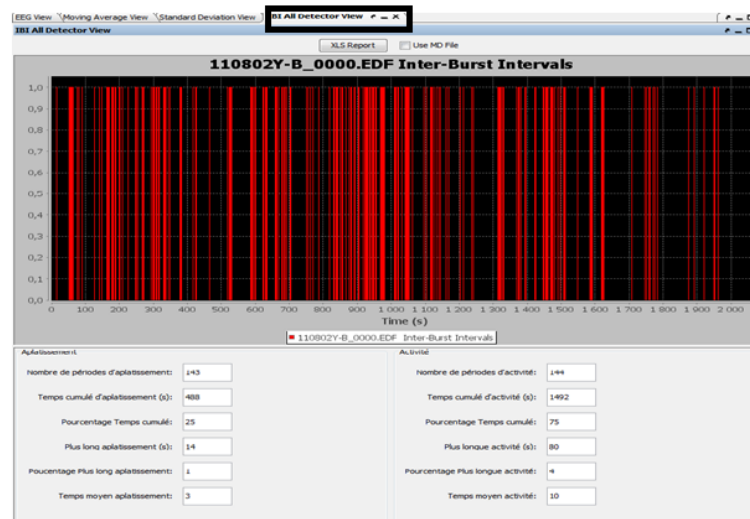


Figure 6: Fenêtre de résultat de la détection des IBI, qui complète le surlignage des IBI sur l'affichage du signal EEG ; en haut : chaque barre rouge correspond à un IBI détecté ; en bas : différentes statistiques sur les IBI (nombre, longueur max., moyenne,...) et les Bursts

Le module *IBI EEG Detector* produit une fenêtre (voir figure 6) avec un graphique de type code barre dans la partie supérieure (une barre est un IBI) et diverses statistiques sur les IBI et les phases d'activité dans la partie inférieure. Nous avons étudié plus spécifiquement 4 analyseurs de détection des IBI pour différentes valeurs du seuil V_T . Chaque analyseur est décrit par sa séquence d'exécution :

- analyseur *SVTF* ("Standard deviation Variation Threshold Filtered") : module *Filter* - module *Smoother* - module *Standard deviation* - module *Standard deviation Variation Threshold* - module *IBI EEG detector*;
- analyseur *SVT* ("Standard deviation Variation Threshold") : idem *SVTF*, sans *Filter*;

- analyseur *STF* ("Standard deviation Threshold Filtered") : module *Filter* - module *Smoother* - module *Standard deviation* - module *Standard deviation Threshold* - module *IBI EEG detector*;
- analyseur *ST* ("Standard deviation Threshold") : idem *STF*, sans le module *Filter*.

Nous avons utilisé pour évaluer la qualité de ces analyseurs et le paramètre de seuil (le plus critique parmi tous les paramètres) à la fois une analyse visuelle du résultat sur quelques EEG, ainsi qu'une méthode inspirée de l'analyse des données sur une base de 408 EEG préalablement classés par le praticien en "normal", "douteux" et "pathologique". Le paragraphe suivant présente succinctement la méthode et les premiers résultats.

5 Une évaluation de la qualité des analyseurs de détection des IBI

Chaque EEG est classé "normal", "douteux" ou "pathologique" (classes nommées par la suite A, B et C respectivement). Nous avons vu en introduction que les statistiques sur les IBI sont de bons indicateurs d'un développement anormal de l'enfant. Nous allons donc étudier la classification des EEG selon les trois variables normalisées suivantes: la durée de l'IBI le plus long l , la durée moyenne des IBI m et la durée totale des IBI t . La normalisation consiste à diviser les 3 variables d'origine par la durée de l'EEG, tous les EEG n'étant pas de la même longueur (de 25 à 35 mn). Ainsi nous pouvons associer chaque EEG à un point libellé de coordonnée (l, m, t) dans $[0,1] \times [0,1] \times [0,1]$, où le label est sa classe (A, B ou C). Nous avons donc un ensemble de n points répartis dans 3 clusters A, B et C . Le choix de l'analyseur et de ses paramètres déplace les points (l, m, t) , déformant ces 3 clusters.

Nous proposons une évaluation de la qualité d'un analyseur basée sur sa capacité à produire des clusters se recouvrant le moins possible. Nous cherchons l'analyseur (pour un paramétrage donné) qui minimise l'inertie intra-classe (les points appartenant à un cluster sont les moins dispersés possibles) et qui maximise l'inertie inter-classe (les centres de chaque cluster sont les plus éloignés possibles les uns des autres). Avec le produit scalaire standard, l'inertie totale est définie par:

$$I_{tot} = \sum_{i=1}^n \langle x_i - c_{A \cup B \cup C}, x_i - c_{A \cup B \cup C} \rangle \quad (1)$$

et l'inertie d'un cluster K (avec $K=A, B$ or C) est donnée par $I_K = \sum_{i=1}^n \langle x_i - c_K, x_i - c_K \rangle$, où x_i est le i^{ieme} point de l'ensemble, $c_{A \cup B \cup C}$ son centre et c_K le centre du cluster K . Le théorème de Huygens nous dit que l'inertie totale de l'ensemble est la somme de l'inertie intra-classe et de l'inertie inter-classe I_{inter} :

$$I_{inter} = \sum_{i=1}^n \langle x_i - c_{A \cup B \cup C}, x_i - c_{A \cup B \cup C} \rangle - I_A - I_B - I_C \quad (2)$$

Nous obtenons $I_{inter} = d_{AB} + d_{AC} + d_{CB}$ où d_{KL} , appelée distance de Ward entre les clusters K et L ($K, L \in \{A, B, C\}$), est définie par:

$$d_{KL} = \frac{n_K n_L}{n} \langle c_K - c_L, c_K - c_L \rangle \quad (3)$$

n_K et n_L étant les nombres de points dans les clusters K et L respectivement.

Nous avons calculé pour les 4 analyseurs *ST*, *STF*, *SVT* et *SVTF* les distances de Ward et les inerties intra-classes pour le seuil V_T variant de 5 à 100 μV (voir figure 7). Nous avons d'abord constaté que l'ajout du filtre à 50Hz n'avait que peu d'influence sur la distance de Ward entre les clusters A (EEG non pathologiques) et C (EEG pathologiques) ainsi que sur leurs inerties. De

plus ce filtre représente environ 30% du temps de calcul total des analyseurs *STF* et *SVTF*, pour un gain finalement très faible en terme de discrimination: la moyenne mobile (très rapide) est suffisante, et nous nous concentrons sur les analyseurs *ST* et *SVT*. En observant les courbes pour *SVT* et le détail des tableaux nous pouvons constater que:

- plus la valeur du seuil est faible plus l'inertie est faible, choix de $V_T \leq 15$,
- plus la valeur du seuil est proche de 20 plus la distance de Ward est grande, choix de $V_T \geq 13$.

Ces résultats sont corroborés par l'analyse visuelle sur quelques EEG: le seuil de $13 \mu V$ fournit les IBI les plus proches de ceux déterminés par le praticien. Pour l'analyseur *ST* nous avons constaté que plus la valeur du seuil est faible plus l'inertie est faible (choix de $V_T \leq 40$) et plus la valeur du seuil est proche de 40 plus la distance de Ward est grande (choix de $V_T \geq 30$), corroborant un choix visuel d'environ $40 \mu V$. Enfin, le temps de calcul par EEGDiag des IBI sur les 408 EEG par l'analyseur *ST* est de 19mn48s et par l'analyseur *SVT* de 19mn et 39s. les 408 EEG faisant une durée totale d'environ 12240mn, nous avons une durée d'analyse environ 644 fois plus rapide que la durée des EEG, ce qui répond bien à notre l'objectif d'une utilisation en routine.

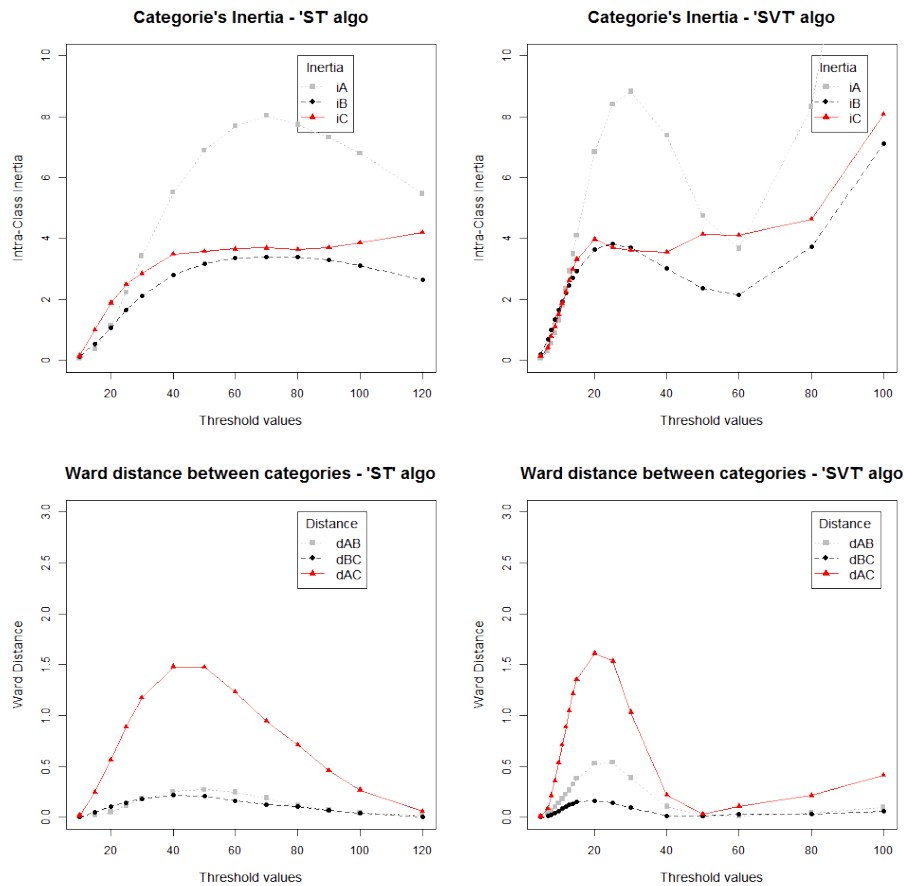


Figure 7: Distances de Ward et inerties intra-classes pour les analyseurs *ST* et *SVT*, pour des valeurs du seuil de détection V_T variant de 5 à $100 \mu V$; iA, iB et iC sont respectivement les inerties intra-classes des clusters A, B et C, et dAB, dBC et dAC sont les distances de Ward entre ces clusters

6 Conclusion

Seuls 21 modules ont été implémentés pour EEGDiag, essentiellement en lien avec la détection des IBI et le filtrage (détection des bandes Alpha, Beta, Delta, etc.). Nous sommes en train d'enrichir le logiciel avec des modules pour l'analyse du risque (complexité de Lempel-Ziv, réseaux neuronaux) en utilisant des données sur le devenir de l'enfant. Il existe des travaux qui utilisent l'analyse visuelle des IBI pour corroborer les résultats de l'analyse automatique (voir par exemple [7]), mais cela n'est possible que sur quelques EEG. Comme nous l'avons montré dans le paragraphe 5, il est possible de construire des outils d'analyse de la qualité sur des grands corpus d'EEG, à condition que l'analyse automatique de l'EEG repose sur des algorithmes rapides.

Enfin, dans le projet BBEEG initial, les outils d'analyse du signal étaient positionnés principalement comme une aide pour les praticiens. Au fur et à mesure du projet, nous avons vu se dégager un intérêt supplémentaire de la plate forme comme support d'enseignement et de recherche pour les étudiants ingénieurs, qui peuvent utiliser le corpus d'EEG comme données sources pour développer de nouvelles approches. Cela nous incite à implémenter des cours spécifiques pour les élèves ingénieurs sur la plateforme d'E-learning.

References

- [1] André M, Fresson J, Ancel P.Y, Lamblin M.D, Hascoët J.M et le groupe EPIPAGE, *Devenir à 5 ans des enfants grands prématurés de la cohorte EPIPAGE : apport de l'EEG de la période néonatale*, Journées Francophones de Recherche en Néonatalogie, Paris (France), 2009.
- [2] E. Biagioni, MF. Frisone, S. Laroche, BA. Kapetanakis, D. Ricci, M. Adeyi-Obe M et al., *Maturation of cerebral electrical activity and development of cortical folding in young very preterm infants*, Clin Neurophysiol, J. PubMed, vol.118(1), pp.53-9, Jan. 2007.
- [3] Watanabe K, Hayakawa F, Okumura, *Neonatal EEG: A powerful tool in the assessment of brain damage in preterm infants*, Brain Dev 21:361- 372, 1999.
- [4] Nguyen The Tich S, Vecchierrini MF, Debillon T, Pereon Y, *Effects of sufentanil on electroencephalogram in very and extremely preterm neonates*, Pediatrics 2003.111:123-128.
- [5] Ralf Biedert, Nicolas Delsaux et al., Java Simple Plugin Framework, <https://code.google.com/p/jspf/>, 2011.
- [6] Illya Yalovyy et al. , Better Swing Application Framework, <https://kenai.com/projects/bsaf/pages/Home>, 2011.
- [7] Ninah Koolen, Katrien Jansen, Jan Vervisch, Vladimir Matic, Maarten De Vos, Gunnar Naulaers and Sabine Van Huffel, *Automatic Burst Detection based on Line Length in the Premature EEG*, BIOSTEC 2013 (Barcelone, Espagne), 11-14 février 2013.

Remerciements

Ces travaux ont été menés avec le support de l'Agence Nationale de la Recherche, via le contrat TECSAN 2010 "BB EEG Platform". Les encouragements et idées de l'équipe "Signal, Image et Science de la Vie" du LISA et des collègues du laboratoire GRIT (Université Libanaise, IUT de Saïda) furent précieux.