



Département
de Physique
École normale
supérieure



Détection et Classification de Signaux Neuronaux

Stage Expérimental

- *Encadrant* : Jérémy Cheslet

- *Maître de Stage* : Timothée Levi

Équipe TIPS

Andrea Combette

RÉSUMÉ

Le traitement d'un signal neuronal permet de remonter à une grande quantité d'informations. En effet, connaître quel neurone est excité à un instant donné, c'est pouvoir comprendre dans une certaine mesure le langage de notre cerveau. La classification des signaux neuronaux est donc un domaine privilégié des neurosciences, permettant à travers l'analyse en temps réel des informations cérébrales de remonter aux actions motrices envisagées par le sujet. Une telle analyse peut être mise en place en utilisant un réseau de neurones artificiels. Le but de ce stage a donc été de créer des données biologiquement réalistes pour tester le réseau, de créer une interface graphique pour ce dernier et de l'optimiser.

The processing of a neural signal can provide a wealth of information. Knowing which neuron is excited at a given moment, we can understand to a certain extent the language of our brain. The classification of neuronal signals is therefore a key area of neuroscience, enabling us to real-time analysis of cerebral information to trace the motor actions envisaged by the subject. One example is the provision of neuro-connected prostheses for tetraplegic patients. Such an analysis can be implemented using an artificial neural network. The aim of this internship was therefore to create biologically realistic data to test the network, create a graphical interface for it and optimize it.

ABSTRACT

Table des matières

I Présentation du réseau de neurones	3
1 Caractéristiques du Réseau	3
a Spiking Neural Network	3
b Règles de plasticités et d'inhibitions	3
2 Structure du réseau	3
a Couche d'entrée	4
b Couche intermédiaire	4
c Couche de sortie	4
3 Objectifs du stage	4
a Générer des signaux neuronaux par simulation	4
b Mise en place d'une interface graphique pour le réseau	4
c Test de l'efficacité du réseau pour plusieurs SNR	4
II Simulation d'une culture de neurones	5
1 Mise en place des simulations	6
a Choix des paramètres et du système	6
b Choix des électrodes	6
c Calibrage du SNR	6
2 Outils d'Analyse des signaux obtenus	6
a Table de vérité	6
b Formes des signaux aux électrodes	6
c Caractéristiques des signaux neuronaux	6
III Travail sur le Spike-Sorting	7
1 Création d'une interface graphique pour la simulation et le réseau	7
a Choix des paramètres du réseau	7
b Observation des données en temps réel	7
c Mise en parallèle du code	7
2 Résultats sur les simulations de cultures	8
a Résultats pour plusieurs SNR	8
b Comparaisons aux données d'origine	8
3 Études de différents paramètres sur l'efficacité à bas SNR	8
a Couche d'entrée : $N_{overlap}$ et ΔvM	8
IV Conclusion	9

I Présentation du réseau de neurones

1 Caractéristiques du Réseau

a Spiking Neural Network

Le réseau de neurone utilisé pour la reconnaissance des potentiels d'action est un Spiking Neural Network (SNN). C'est un réseau de neurones dynamiques qui contraste avec les réseaux de neurones traditionnels, il permet d'ajouter une dimension temporelle à l'apprentissage [1]. En effet, l'apprentissage d'un neurone ne va pas être uniquement fonction des stimuli qu'il reçoit à un instant, mais aussi des stimuli précédents, à travers la variation de potentiel de leur membrane. Ceci est biologiquement plus réaliste et moins demandant en ressource, surtout lors de l'apprentissage [5]. L'idée est donc d'utiliser ce type de réseau pour traiter le problème de classification évoqué.

b Règles de plasticités et d'inhibitions

Le SNN utilisé présente plusieurs couches de neurones et donc des synapses, entre un neurone pré et postsynaptique. Ces dernières se caractérisent par une certaine plasticité (Spike-Timing-Dependent Plasticity, **STDP**), renforçant l'aspect bio-mimétique du réseau. Cette plasticité s'exprime par une dépendance du poids w_i de la synapse en fonction de l'intervalle de temps entre le stimulus du neurone postsynaptique et celui du neurone présynaptique. Pour une plasticité additive [1], le poids d'une synapse varie de $\pm \Delta w$ selon que le neurone postsynaptique soit ou non excité en même temps que le neurone présynaptique (à une fenêtre de temps près). Ce type de synapse présente une mémoire longue puisque si le couple neurone pré et postsynaptique ne sont plus coordonnés dans leur excitation, le poids de la synapse reste invariant. Cette spécificité permet un apprentissage non supervisé du réseau (les données ne sont pas étiquetées) [4]

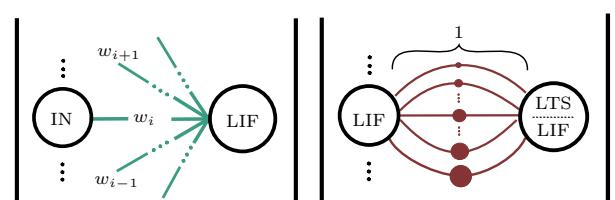
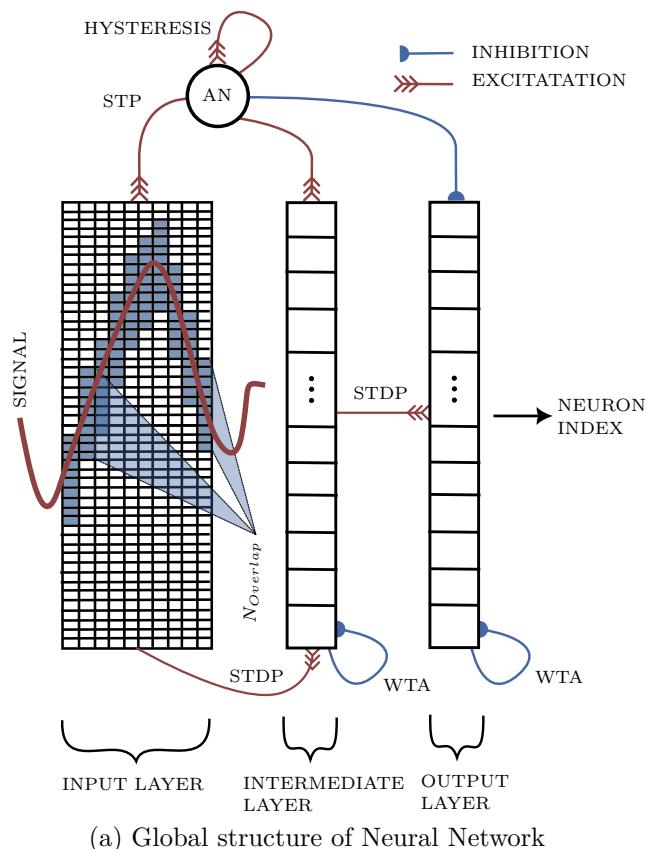
Dans certains cas, la contrainte dynamique est telle que si le poids de la synapse reste inchangé l'apprentissage est diminué. Il est donc nécessaire d'introduire une capacité d'oubli dans certaines synapses. On parle de synapse **STP** (Short Term Plasticity). Ainsi si les deux neu-

rones pré-post-synaptique se découpent dans leur excitation la synapse voit son poids diminué. Cette dépendance étant traduite par l'équation suivante :

$$\frac{dw}{dt} = \frac{1}{\tau_{STP}}(1 - w) - w.f_d \sum_s \delta(t - t_s)$$

Un autre mécanisme peut être mis en place, celui du "Winner Takes All" (WTA), ainsi lorsqu'un neurone est excité dans une couche, il inhibe les neurones de la même couche. Cela permet entre autres de réservoir l'apprentissage d'une forme de potentiel d'action à un unique neurone.

2 Structure du réseau



(b) Input layer synapses (c) output layer synapses

FIGURE 1 – Structure du réseau et des synapses

1. Synapses with different delays (see Fig.1.c)

a Couche d'entrée

La couche d'entrée du réseau est composée de neurones "sensoriels", ils permettent en quelque sorte de prendre une photo du signal d'entrée. Ils sont répartis, comme on le voit dans la Fig.1, selon une grille de dimension $(1/\Delta V_M + 2N_{overlap}, 10)$, où ΔV_M est la sensibilité d'un neurone de la couche, et $N_{overlap}$ le nombre de neurones qui peuvent s'activer sur une même colonne (effet filtre passe bas). On notera que ΔV_M est dépendant linéairement du bruit du signal. Pour chaque colonne de la couche d'entrée, l'activation est retardée de 0.05 ms, on peut donc "photographier" 0.5 ms de signal. Or un potentiel d'action a une durée moyenne de quelques ms, Il faudra donc plusieurs "photographies" pour reconstituer un pic (rôle de la couche de sortie).

Enfin, elle est connectée à la couche intermédiaire par des synapses excitatrices STDP. Ainsi chaque neurone de la couche intermédiaire est connecté à tous les neurones d'entrée (Fig 1.b), ils perçoivent donc chacun le signal d'entrée à travers la couche sensorielle. Mais ceci de façon différente. En effet, le poids des synapses va varier d'un neurone de la couche intermédiaire à l'autre.

b Couche intermédiaire

La couche intermédiaire est composé de neurones **LIF** (Leak Integrate and Fire). C'est ce type de neurone qui donne toute sa spécificité au SNN [1]. Chaque neurone de cette couche a pour objectif de mémoriser un fragment de potentiel d'action et d'être excité à chaque fois que ce motif apparaît dans le signal d'entrée. Pour éviter que le neurone mémorise du bruit, le réseau de neurones intègre un neurone d'attention (A.N) qui joue dans un rôle crucial dans l'ensemble du réseau. Ce neurone est un neurone LIF, qui agit comme détecteur de potentiel d'action. Dès que le signal va dépasser un certain seuil (Threshold), le neurone d'attention va exciter les neurones de la couche intermédiaire, permettant ainsi la mémorisation du potentiel d'action.

Comme le montre la Fig.2.c, la couche intermédiaire est ensuite connectée à la couche de sortie par des synapses STDP. Cette dernière couche a pour but de rassembler les différents fragments de potentiel d'action et de les attribuer à un neurone.

c Couche de sortie

La couche de sortie quant à elle, permet la classification des signaux neuronaux. Ce sont ici soit des neurones LTS (Low Threshold Spiking) ou des neurones LIFs légèrement modifiés [1], qui sont connectés à tous les neurones de la couche intermédiaire. Ainsi, si plusieurs neurones de la couche intermédiaire sont excités dans un intervalle de temps réduit, échantillonnant un seul potentiel d'action, ces neurones vont entraîner l'excitation d'un neurone de la couche de sortie. On aura alors un seul neurone excité dans cette couche (action du WTA), qui encodera l'ensemble des fragments de potentiel d'action issus de la couche précédente.

3 Objectifs du stage

a Générer des signaux neuronaux par simulation

L'étude de l'efficacité du réseau a précédemment été faite sur des signaux artificiels de la forme [1] :

$$V(t) = A \cos(2\pi \frac{t - t_{ph}}{\tau_1}) \exp(-(\frac{2.3548t}{\tau_2})^2)$$

Le premier objectif de ce stage a donc été de générer des données biologiquement plus réalistes, en s'aidant d'une bibliothèque python **MEA-Rec**. Il a fallu aussi fournir les outils nécessaires à la caractérisation des données et à leur implémentation dans le réseau de neurone.

b Mise en place d'une interface graphique pour le réseau

Le deuxième objectif du stage a été de mettre en place une interface graphique pour régler les paramètres du réseau et surtout pour suivre en temps réel l'apprentissage du réseau. Ceci afin de faciliter la prise en main de l'outil par les biologistes.

c Test de l'efficacité du réseau pour plusieurs SNR

Enfin, il fallait tester l'efficacité du réseau sur plusieurs signaux générés et de rapport signal

sur bruit différents (SNR). Le SNR étant défini dans notre étude de la manière suivante :

$$SNR = \frac{\langle A_{max} \rangle}{\sigma_{noise}},$$

Où A_{max} représente l'amplitude d'un potentiel d'action et σ_{noise} l'écart type du bruit. On s'at-

tend donc à ce que l'apprentissage soit moins fructueux pour des SNR bas, les signaux étant plus bruités.

II Simulation d'une culture de neurones

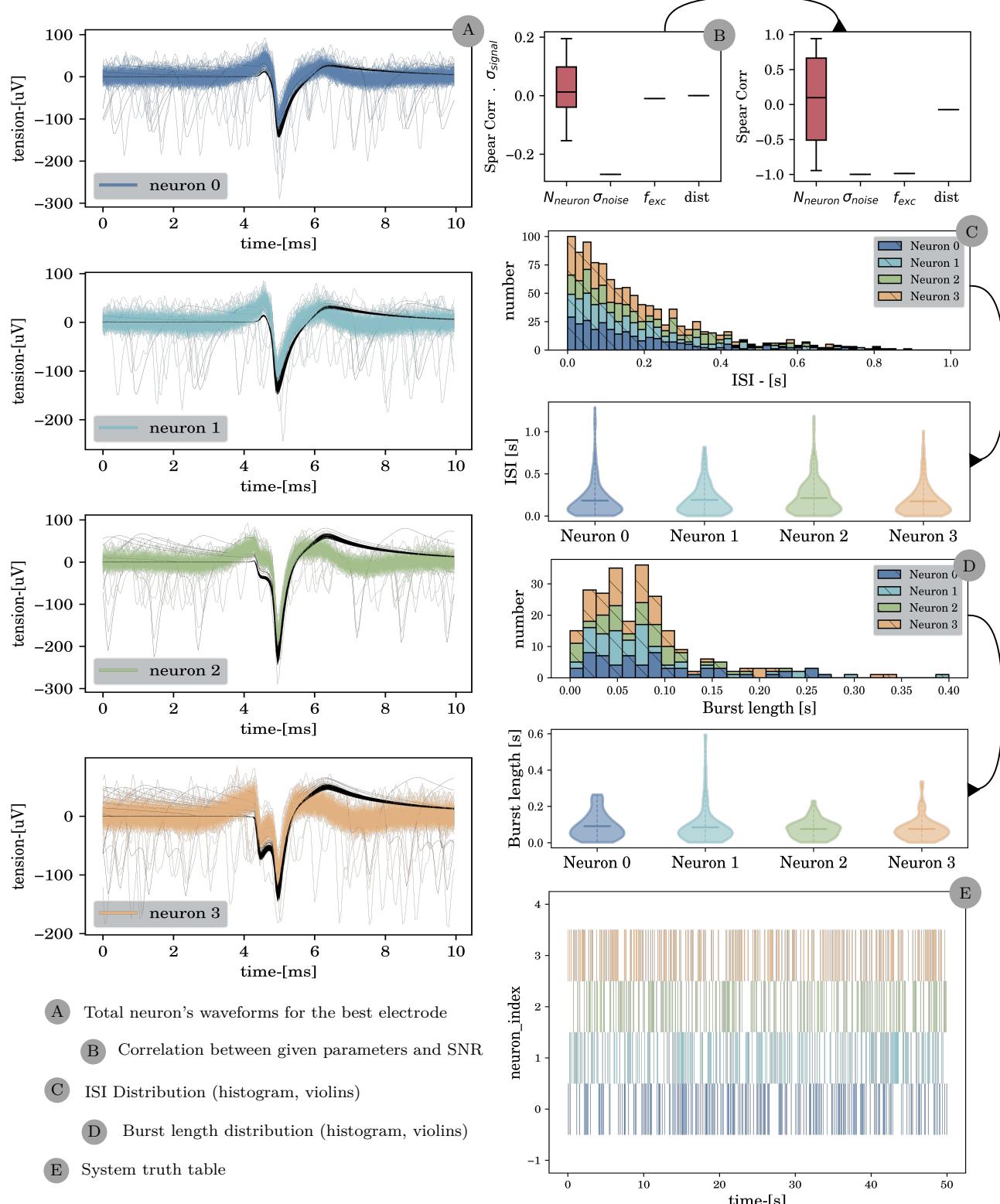


FIGURE 2 – Différentes figures issues des simulations

1 Mise en place des simulations

a Choix des paramètres et du système

MEARec permet de modéliser un système composé d'une ou plusieurs électrodes et de neurones qui se disposent dans les 3 dimensions autour de l'électrode. Certains paramètres de cette modélisation peuvent être modifiés, comme l'écart type du bruit, l'intervalle de variation de l'amplitude des potentiels d'action, le type de neurones, leur nombre, leur fréquence d'excitation et leur distance à l'électrode ... Le but final étant de générer plusieurs signaux avec des SNR différents. Il m'a fallu étudier l'impact des différents paramètres sur ce dernier. Après plusieurs simulations, il apparaît que seul l'écart type du bruit et l'intervalle de variation de l'amplitude des potentiels d'action aient une influence sur le SNR. En effet, comme le montre le deuxième graphique de la Fig.2.B, le coefficient de corrélation de Spearman est non nul pour l'écart type du bruit et pour la fréquence d'excitation. Cependant la dépendance du SNR en fréquence est anecdotique, comme le montre le premier graphique de la Fig.2.B.

b Choix des électrodes

Pour l'étude du réseau de neurone, il est apparu qu'il était plus simple de travailler avec une seule électrode, la structure finale n'étant pas prête pour la mise en parallèle de plusieurs couches d'entrée et de couches intermédiaires nécessaire au fonctionnement du SNN avec plusieurs électrodes [1]. Il a tout de même fallu écrire un code pour générer les électrodes désirées pour de futures études.

c Calibrage du SNR

Pour l'étude de l'efficacité du réseau en fonction du SNR, j'ai décidé de fixer l'intervalle de variation des potentiels d'action dans une fenêtre de $[50, 300] \mu\text{v}$ et de faire varier l'écart type du bruit. Pour générer des données à SNR donné, il a donc fallu modéliser l'évolution du SNR par rapport à l'écart type du bruit pour le système étudié. C'est l'étape d'étalonnage du système pour pouvoir étudier l'impact du SNR sur le réseau. Ceci revient à prédire pour le système donné, la valeur de $\langle A_{max} \rangle$. Cette étape est réalisée avec un simple fit en $\frac{a}{x}$.

2 Outils d'Analyse des signaux obtenus

a Table de vérité

L'évaluation de l'efficacité du réseau nécessite une table de vérité de l'enregistrement. Ce tableau contient la localisation temporelle de chaque potentiel d'action pour chaque neurone. Cette évaluation se fait à travers trois paramètres : la **Précision** (\mathcal{P}), le **Recall** (\mathcal{R}) et le **F-Score** (\mathcal{F}) ou moyenne harmonique.

$$\mathcal{P} = \frac{TP}{TP + FP}$$

$$\mathcal{R} = \frac{TP}{TP + FN}$$

$$\mathcal{F} = \frac{TP}{N_{Tot} + N_{Det}}$$

Ici TP désigne le nombre de vrai positif, FN le nombre de faux négatif, FP le nombre de faux positif, N_{Tot} le nombre de pics d'après la table de vérité, et N_{Det} le nombre de pics détectés.

b Formes des signaux aux électrodes

Pour comparer les motifs appris par le réseau aux potentiels d'action réels, il a fallu superposer tous les signaux de chaque neurone à la meilleure électrode (SNR le plus élevé) tout au long de l'enregistrement. Comme nous pouvons le voir dans la Fig.2.A, les potentiels d'action ont des formes relativement similaires avec des caractéristiques que le réseau de neurone devrait pouvoir isoler. Le but final de ce travail étant de voir quel neurone de la couche de sortie a appris à quel neurone de la culture (à travers une convolution des 2 signaux neuronaux).

c Caractéristiques des signaux neuro-naux

Pour caractériser le signal reçu par la culture simulée, plusieurs outils ont été mis à disposition. Comme le calcul de l'**ISI** (distribution de l'intervalle de temps entre les spikes), de l'**ISI** (distribution de l'intervalle de temps entre les bursts) et de la **durée des bursts** [2], comme nous pouvons le voir avec la Fig.2.C et Fig.2.D, où la distribution des différents paramètres semble suivre une loi de Poisson. Ceci est cohérent avec les paramètres d'entrée des simulations, puisque les potentiels d'action suivent un processus Poissonien.

III Travail sur le Spike-Sorting

1 Création d'une interface graphique pour la simulation et le réseau



FIGURE 3 – Interface graphique

a Choix des paramètres du réseau

Pour cette interface graphique il m'a été demandé de créer un onglet où l'on pourrait régler les différents paramètres du réseau, les caractéristiques des couches ... Une fois la configuration faite l'utilisateur peut lancer la simulation et suivre l'apprentissage du réseau grâce aux 3 autres onglets.

b Observation des données en temps réel

Un suivi en temps réel des poids des STDP entre les neurones sensoriels et les LIF intermédiaire a été mis en place, permettant de voir quel motif est appris. Il est également possible de voir le signal brut et d'observer l'état des différents neurones de la couche intermédiaire et de la couche finale, c'est à dire quand ils sont excités. On peut également suivre la forme des signaux à l'électrode, appris par la couche finale, ainsi que la correspondance entre les patterns appris et les vrais patterns de potentiel d'action. Un dernier graphe permet de connaître le rapport du

nombre de pics détectés pour chaque neurone reconnu, ce pour tous les neurones de la couche finale. Ces graphiques permettent de vérifier si l'apprentissage est correct pour tous les neurones et si ce n'est pas le cas, ils permettent de discriminer le ou les neurones qui posent problème

c Mise en parallèle du code

Pour gérer autant de tâche en même temps sans ralentir considérablement la simulation, il a fallu mettre en parallèle les différentes parties du programme. Ceci, en ayant recours au **QThreads** [3] qui ont permis d'accélérer considérablement le code. Avec un Thread dédié à chaque groupe de graphique, la vitesse d'exécution de la simulation reste presque inchangée par rapport à la version du code sans interface graphique. On a en effet, une diminution de 1000 **itérations** sur 1100, ce qui reste raisonnable par rapport aux modifications apportées. Cette interface constitue donc une belle avancée dans l'accessibilité du projet.

2 Résultats sur les simulations de cultures

a Résultats pour plusieurs SNR

Comme prévu, le SNR joue un rôle non négligeable sur les performances du réseau. Cependant, comme nous pouvons le voir dans la Fig.4 le SNR a un impact varié dans les performances des différentes couches.

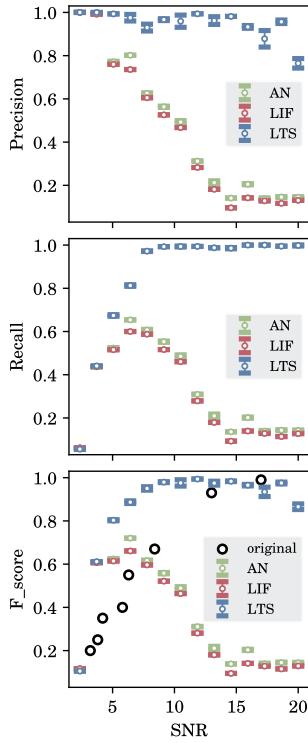


FIGURE 4 – Performance du réseau en fonction du SNR

ce qui implique que les pics issus du bruit peuvent être détectés par les deux premières couches.

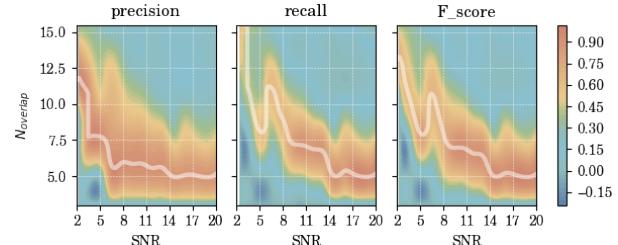
b Comparaisons aux données d'origine

Comme le montre la Fig.4, les données d'origine (Partie 3.a) présentent étonnamment des résultats de moins bonne qualité à bas SNR, alors que les données sont beaucoup plus régulières. L'écart en F_{score} entre les données d'origine et les données simulées est en effet de 0.2 à 0.4, il est donc probable que le réseau ait de bonnes performances avec de vraies cultures de neurones.

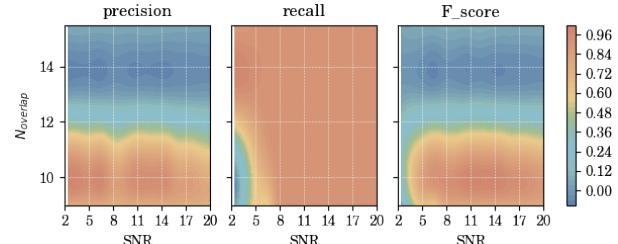
3 Études de différents paramètres sur l'efficacité à bas SNR

On peut alors chercher à trouver les paramètres optimaux de la couche d'entrée, qui conditionne la qualité de la "photographie" du signal.

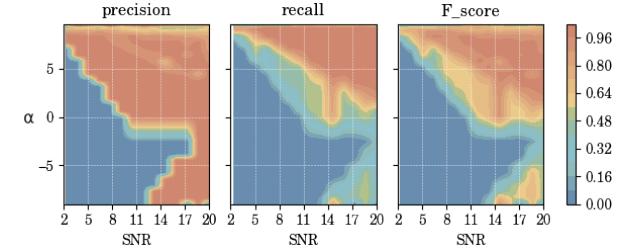
a Couche d'entrée : $N_{overlap}$ et ΔvM



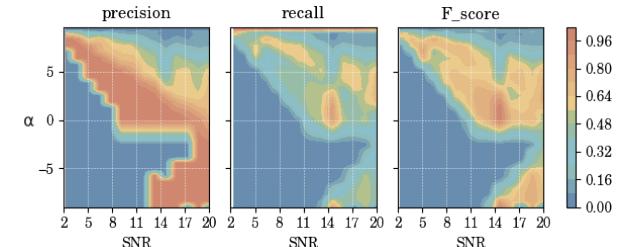
(a) évolution pour le $N_{overlap}$, à la couche intermédiaire



(b) évolution pour le $N_{overlap}$, à la couche finale



(c) évolution pour le coefficient de sensibilité, à la couche intermédiaire



(d) évolution pour le coefficient de sensibilité, à la couche finale

FIGURE 5 – Evolution des scores pour différents paramètres

On rappelle que la sensibilité des neurones ΔV_M est dépendante linéairement de l'écart type du bruit. On a donc fait varier le coefficient de sensibilité de la façon suivante : $\{(10 - \alpha), \alpha \in$

$[-10; 10]\}$.

Comme nous pouvons le voir sur la Fig.5.a, un grand $N_{overlap}$ permet de bons résultats à faible SNR pour la couche intermédiaire, ce qui est cohérent puisque ce paramètre agit comme un filtre passe-bas, alors que pour des SNR plus élevés, c'est typiquement l'inverse. Or comme le montre la Fig.2.b il n'y a une détection par la couche finale uniquement si le $N_{overlap}$ est supérieur à 9, on ne pourra donc pas descendre le $N_{overlap}$ aussi loin que nous le voudrions pour des SNR hauts. Ainsi, dans la plupart des cas on devra utiliser un $N_{overlap}$ de l'ordre de la dizaine, les faux positifs étant éliminés par la couche de sortie pour des SNR élevés (Fig.2.b).

Concernant le coefficient de sensibilité des neurones sensoriels, son impact est inverse à celui du $N_{overlap}$ pour les deux couches (Fig.5.c et .d). En effet, plus le coefficient de sensibilité est faible plus l'image faite par la couche d'entrée est précise. On aura donc avantage, à faible SNR, d'utiliser un grand $N_{overlap}$ et une petite sensibilité, dans la mesure des ressources en calcul disponible (puisque le nombre de neurones à traiter augmente).

IV Conclusion

Ce stage à l'interface entre les neurosciences et l'informatique a été une introduction au spike sorting et à de nombreux outils Python. Ces derniers ont permis de mettre au point une procédure de génération de signaux, une interface graphique en passant par l'optimisation de certains paramètres du réseau pour des signaux bruités. Pour aller plus loin dans cette étude il faudrait étudier d'autre type de dépendance que la linéarité en bruit de ΔV_M . L'implémentation de synapses dans nos simulations serait aussi un point important à traiter, ceci serait bien plus précis et fidèle d'un point de vue biologique. En effet l'analyse des signaux issus des synapses apporteraient des éléments discriminatifs supplémentaires pour l'étape de tri.

Plus généralement, cette méthode de classification ouvre le fait qu'elle soit non supervisée pré-

sente un grand intérêt scientifique, elle est générale. Ainsi tout signal numérique qu'il soit d'origine biologique ou non, peut être analysé et classifié (en réglant les paramètres du réseau en fonction des temps caractéristiques des phénomènes étudiés). Par exemple en physique, la détection de forte variation de pression dans un écoulement et la caractérisation de turbulences en microfluidique pourraient être des domaines d'application, la taille et le temps propres des tourbillons étant accessibles à travers la simple exploitation des résultats du réseau. Des applications plus directes apparaissent dans le domaine médical où la mise à disposition de prothèses neuro-connectées permettraient à des patients tétraplégiques de retrouver un usage plus naturel de ces dernières.

Références

- [1] M. BERNERT. In : *Development* (2019). URL : <https://tel.archives-ouvertes.fr/tel-03035855>.
- [2] L. L. BOLOGNA et al. “Investigating neuronal activity by SPYCODE multi-channel data analyzer.” In : *Neural networks : the official journal of the International Neural Network Society* 23.6 (2010), p. 685-697. DOI : 10.1016/j.neunet.2010.05.002. URL : <https://pubmed.ncbi.nlm.nih.gov/20554151>.
- [3] Qt. COMPANY. “QThreads Documentation and Public Function”. In : (). URL : <https://doc.qt.io/qt-5/qthread.html>.
- [4] T. MASQUELIER, R. GUYONNEAU et S. J. THORPE. “Competitive STDP-based spike pattern learning.” In : *Neural Comput.* 21.5 (2009), p. 1259-1276. DOI : 10.1162/neco.2008.06-08-804. URL : <https://pubmed.ncbi.nlm.nih.gov/19718815>.
- [5] A. MEHONIC et A. J. KENYON. “Emulating the Electrical Activity of the Neuron Using a Silicon Oxide RRAM Cell.” In : *Frontiers in neuroscience* 10 (2016), p. 57. DOI : 10.3389/fnins.2016.00057. URL : <https://pubmed.ncbi.nlm.nih.gov/26941598>.