



**HAL**  
open science

# Estimation de paramètres de modèles de neurones biologiques sur une plate-forme de SNN (Spiking Neural Network) implantés "in silico"

Laure Buhry

► **To cite this version:**

Laure Buhry. Estimation de paramètres de modèles de neurones biologiques sur une plate-forme de SNN (Spiking Neural Network) implantés "in silico". Micro et nanotechnologies/Microélectronique. Université Sciences et Technologies - Bordeaux I, 2010. Français. NNT : . tel-00561396

**HAL Id: tel-00561396**

**<https://theses.hal.science/tel-00561396v1>**

Submitted on 1 Feb 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# THÈSE

PRÉSENTÉE À

**L'UNIVERSITÉ DE BORDEAUX I**

ÉCOLE DOCTORALE DE SCIENCES PHYSIQUE ET DE  
L'INGÉNIEUR

Par **Laure Buhry**

POUR OBTENIR LE GRADE DE

**DOCTEUR**

EN SCIENCES

SPÉCIALITÉ : ÉLECTRONIQUE

---

**Estimation de paramètres de modèles de neurones  
biologiques sur une plate-forme de SNN (Spiking  
Neural Network) implantés "in silico"**

---

Soutenue le : 21 Septembre 2010

**Après avis des rapporteurs :**

Frédéric ALEXANDRE Directeur de Recherche INRIA

Gérard DREYFUS .... Professeur ESPCI-Paris Tech

**Devant la commission d'examen composée de :**

Alain DESTEXHE .... Directeur de Recherche CNRS Président

Frédéric ALEXANDRE Directeur de Recherche INRIA Rapporteur

Gérard DREYFUS .... Professeur ESPCI-Paris Tech . Rapporteur

Eric GRIVEL ..... MCF IPB ..... Examineur

Sylvie RENAUD ..... Professeur IPB ..... Directrice de thèse

Sylvain SAÏGHI ..... MCF Université Bordeaux 1 .. Co-directeur de Thèse



# Remerciements

Cher lecteur, c'est par cette page que j'achève la rédaction de mon manuscrit. Je suis heureuse de pouvoir ainsi exprimer ma sincère gratitude à tous ceux qui m'ont accompagnée pendant mes trois années de doctorat pour mener à bien ces travaux.

Je remercie tout d'abord Gérard Dreyfus d'avoir accepté de rapporter sur mes travaux de thèse. Sa relecture minutieuse et la pertinence de ses remarques m'ont permis d'améliorer notamment la pédagogie de mon manuscrit.

Je remercie aussi tout particulièrement Frédéric Alexandre qui, lui aussi, a spontanément accepté d'être rapporteur. Grâce à ses questions pertinentes et constructives, j'ai pu mettre davantage en valeur les contributions apportées par mon travail.

Je tiens à remercier également Alain Destexhe pour l'intérêt qu'il a porté à mes travaux et pour avoir accepté de faire partie de mon jury de thèse. Il a su porté un regard différent sur ce sujet, participant ainsi à consolider sa cohérence dans un contexte pluridisciplinaire.

Comment exprimer ma reconnaissance à Sylvie Renaud qui m'a fait confiance en offrant successivement à la mathématicienne que j'étais, un sujet de stage en master 2 Sciences de la Cognition, puis un sujet de thèse? Elle a pris le risque de m'accueillir dans un laboratoire d'électronique pour travailler sur un sujet en neurosciences computationnelles. Elle m'a initiée à la recherche tout en me laissant une totale liberté dans la conduction de mon travail et le choix de mes collaborateurs; je pense notamment à mon séjour de trois mois à New York, au Courant Institute... Je remercie aussi tout naturellement Sylvain Saïghi pour avoir accepté de co-encadrer ma thèse, pour sa disponibilité et son écoute. C'est aussi, et surtout, grâce à eux et au sujet de recherche qu'ils m'ont proposé que le prix *Pour les Femmes et la Science* de la fondation d'entreprise L'Oréal France - UNESCO - Académie des Sciences m'a été attribué en 2009; j'en remercie d'ailleurs le jury et les organisateurs.

Ce travail ayant été réalisé en collaboration avec l'équipe Signal et Image de l'IMS, je me dois de remercier Audrey Giremus et Eric Grivel qui m'ont guidée dans la jungle des algorithmes d'optimisation et ont ajouté une dimension supplémentaire à la pluridisciplinarité de mon sujet de thèse. Je leur suis particulièrement reconnaissante pour le temps qu'ils ont passé à s'approprier le sujet dans un domaine de recherche qui leur était nouveau.

I am really grateful to John Rinzel for welcoming me at the Courant Institute for Mathematical Sciences / Center for Neural Science thanks to the ACCN program. I have been working there in a stimulating environment and started a very rewarding collaboration.

Il m'est impossible de ne pas remercier l'ensemble des membres ~~de l'équipe ISN~~ des équipes AS2N et Elibio (les Rouges et les Bleus...) pour le travail dans la bonne humeur et les nombreux prétextes trouvés pour se réunir dans la « salle Neurone » autour d'un gââââteau. Je garderai également de très bons souvenirs des discussions (pseudo)-philosophiques, une tasse de thé à la main, autour de la table ronde, ou à la mare aux canards. D'une manière plus générale, je souhaiterais remercier tous mes collègues du laboratoire pour l'agréable ambiance de travail.

Bien évidemment, je voudrais remercier mes amis et ma famille pour m'avoir encouragée tout au long de mes études. J'ai conscience du manque de disponibilité dont j'ai fait preuve ces dernières années et qu'ils ne m'ont, pourtant, jamais reproché. J'ai naturellement une pensée particulière pour mes parents dont l'incontestable soutien m'a permis de faire des études et de les poursuivre jusqu'au doctorat.

Je ne peux pas conclure ces remerciements sans un grand « Merci » à Jérémie, qui a toujours su me redonner confiance dans les instants de doute, pour nos moments de complicité et de bonheur qu'il m'offre chaque jour.

## Résumé

**Mots-clés:** modélisation de neurone biologique, estimation de paramètres, optimisation, métaheuristiques, circuits intégrés analogiques neuromimétiques, réseaux de neurones.

Ces travaux de thèse, réalisés dans une équipe concevant des circuits analogiques neuromimétiques suivant le modèle d'Hodgkin-Huxley, concernent la modélisation de neurones biologiques, plus précisément, l'estimation des paramètres de modèles de neurones. Une première partie de ce manuscrit s'attache à faire le lien entre la modélisation neuronale et l'optimisation. L'accent est mis sur le modèle d'Hodgkin-Huxley pour lequel il existait déjà une méthode d'extraction des paramètres associée à une technique de mesures électrophysiologiques (le voltage-clamp) mais dont les approximations successives rendaient impossible la détermination précise de certains paramètres. Nous proposons dans une seconde partie une méthode alternative d'estimation des paramètres du modèle d'Hodgkin-Huxley s'appuyant sur l'algorithme d'évolution différentielle et qui pallie les limitations de la méthode classique. Cette alternative permet d'estimer conjointement tous les paramètres d'un même canal ionique. Le troisième chapitre est divisé en trois sections. Dans les deux premières, nous appliquons notre nouvelle technique à l'estimation des paramètres du même modèle à partir de données biologiques, puis développons un protocole automatisé de réglage de circuits neuromimétiques, canal ionique par canal ionique. La troisième section présente une méthode d'estimation des paramètres à partir d'enregistrements de la tension de membrane d'un neurone, données dont l'acquisition est plus aisée que celle des courants ioniques. Le quatrième et dernier chapitre, quant à lui, est une ouverture vers l'utilisation de petits réseaux d'une centaine de neurones électroniques : nous réalisons une étude logicielle de l'influence des propriétés intrinsèques de la cellule sur le comportement global du réseau dans le cadre des oscillations gamma.

## Abstract

**Keywords:** biological neuron modeling, parameter estimation, optimization, metaheuristics, neuromimetic analog integrated circuits, spiking neural networks.

These works, which were conducted in a research group designing neuromimetic analog integrated circuits based on the Hodgkin-Huxley model, deal with the parameter estimation of biological neuron models. The first part of the manuscript tries to bridge the gap between neuron modeling and optimization. We focus our interest on the Hodgkin-Huxley model because it is used in the group for designing the neuromimetic circuits. There already existed an estimation method associated to the voltage-clamp technique. Nevertheless, this classical estimation method does not allow to extract precisely all parameters of the model, also in the second part, we propose an alternative method to jointly estimate all parameters of one ionic channel, avoiding the usual approximations. This method is based on the differential evolution algorithm. The third chapter is divided into three sections : the first two sections present the application of our new estimation method to two different problems, model fitting from biological data and development of an automated tuning of neuromimetic chips. In the third section, we propose an estimation technique using only membrane voltage recordings – easier to measure than ionic currents. Finally, the fourth and last chapter is a numerical experimentation preparing the implementation of small neural networks on neuromimetic chips. More specifically, we try to study the influence of cellular intrinsic properties on the global behavior of a neural network in the context of gamma oscillations.



# Table des matières

<b>Introduction générale</b>	<b>1</b>
<b>1 Du neurone biologique au modèle</b>	<b>5</b>
1.1 La modélisation de neurones biologiques : de l’anatomie au modèle . . . . .	6
1.1.1 Anatomie et physiologie d’un neurone biologique . . . . .	6
1.1.2 Les modèles mathématiques de la littérature . . . . .	10
1.2 Les méthodes d’optimisation appliquées à l’estimation des paramètres de modèles de neurones : état de l’art . . . . .	14
1.2.1 Quelques principes généraux en optimisation pour l’estimation de paramètres de modèles de neurones . . . . .	15
1.2.2 Application à l’estimation des paramètres de modèles de neurones . . . . .	16
1.3 Retour sur le formalisme d’Hodgkin–Huxley et la méthode du voltage-clamp . . . . .	23
1.3.1 Le formalisme d’Hodgkin–Huxley : une analogie biologie/électricité . . . . .	23
1.3.2 Le voltage-clamp et la méthode disjointe : limitations . . . . .	27
1.4 Résumé . . . . .	35
<b>2 Estimation des paramètres de modèles de neurones à partir des courants ioniques</b>	<b>37</b>
2.1 Le choix de la méthode d’estimation : comparaison de métaheuristiques . . . . .	38
2.1.1 Choix d’une fonction d’erreur . . . . .	38
2.1.2 Protocole expérimental et implémentation des équations du modèle . . . . .	39
2.1.3 Comparaison de méthodes d’optimisation . . . . .	40
2.1.4 Validation numérique de la méthode du DE . . . . .	52
2.2 Amélioration de l’algorithme d’Évolution Différentielle . . . . .	58
2.2.1 Introduction de nouvelles variantes de l’évolution différentielle . . . . .	58
2.2.2 Étude comparative sur un benchmark de fonctions . . . . .	60
2.2.3 Application des variantes au canal potassique du modèle d’Hodgkin–Huxley . . . . .	63
2.3 Résumé . . . . .	65

---

<b>3 Applications</b>	<b>67</b>
3.1 Application à des données biologiques . . . . .	68
3.1.1 Méthode . . . . .	68
3.1.2 Résultats . . . . .	69
3.2 Application au réglage de circuits neuromimétiques . . . . .	70
3.2.1 Méthode . . . . .	70
3.2.2 Résultats – Discussion . . . . .	75
3.3 Estimation des paramètres d’un modèle à partir de la tension de membrane . . . . .	78
3.3.1 Méthode . . . . .	78
3.3.2 Quelques résultats . . . . .	80
3.4 Résumé . . . . .	83
<b>4 Vers la configuration d’un réseau de neurones</b>	<b>85</b>
4.1 Introduction . . . . .	86
4.2 Modèle et méthodes . . . . .	88
4.3 Les différents réseaux de neurones envisagés . . . . .	91
4.3.1 Réseau de neurones excitateurs et inhibiteurs de type I ou II exclusivement	91
4.3.2 Réseau mixte de neurones excitateurs de type I et d’inhibiteurs de type II	100
4.3.3 Réseau de neurones inhibiteurs . . . . .	101
4.4 Conclusion . . . . .	104
4.5 Résumé . . . . .	106

# Introduction générale

Plusieurs disciplines scientifiques se rencontrent depuis quelques décennies autour de problématiques communes aux neurosciences. Ce point de rencontre a donné lieu à un nouveau domaine de recherche appelé « Neurosciences computationnelles ». De nouveaux questionnements sont apparus auxquels les neurosciences computationnelles souhaitent apporter leur éclairage en mêlant le savoir-faire expérimental des biologistes et physiciens, l'abstraction des mathématiciens et informaticiens... Ce regroupement disciplinaire vise en effet à mieux comprendre les relations entre la structure et la fonction cérébrale. Les perspectives en terme d'applications biomédicales, notamment, lui a fait connaître ces dix dernières années un essor considérable.

L'un des moyens privilégiés par cette discipline pour l'étude du fonctionnement du système nerveux est la modélisation : modélisation microscopique à l'échelle de la cellule ou composantes et de ses échanges chimiques, modélisation au niveau mésoscopique de quelques centaines à quelques milliers de neurones, modélisation à plus grande échelle de structures cérébrales (colonnes corticales, ganglions de la base...) ou encore comportements cognitifs. Cependant, tout modélisateur est confronté à cette ambiguïté : comment faire des hypothèses simplificatrices nécessaires à l'élaboration d'un modèle sans perdre la complexité comportementale étudiée ? En outre, plus le modèle sera complexe, plus les ressources calculatoires exigées seront importantes, ce qui limitera par exemple à quelques neurones l'étude de leurs comportements biophysiques microscopiques. A l'inverse, bien que rendant possible la simulation par un ordinateur de l'activité d'un grand ensemble neuronal, la modélisation du comportement global d'un réseau de neurones (comme le modèle de Wilson-Cowan [WC72]) ne nous donnera pas accès à la dynamique individuelle des cellules qui constitueraient son équivalent biologique.

L'ingénierie neuromimétique, dont un des axes de recherche est de concevoir des circuits intégrés analogiques reproduisant l'activité électrique des neurones (le premier circuit fut proposé dans [MD91]), apparaît comme un bon compromis entre simplification et précision. Elle permet entre autre de garder une résolution en temps réel de phénomènes en dépit de leur complexité. Ainsi, il devient possible, par couplage avec un appareillage classique d'électrophysiologie, de relier un tel circuit neuromimétique avec un neurone vivant [RLLMA93], laissant entrevoir une multitude d'applications.

C'est sur la base de ces concepts qu'a été fondée l'équipe Ingénierie des Systèmes Neuromorphiques (ISN) du laboratoire IMS (Intégration du Matériau au Système, UMR 5218 CNRS / IPB-Université Bordeaux 1) au sein de laquelle ces travaux de thèse ont été en grande partie réalisés. Par ailleurs, l'équipe prend place depuis le 1<sup>er</sup> Janvier 2009 dans le groupe Bio-électronique dans le cadre d'un regroupement thématique.

On distinguait parmi les travaux de l'équipe deux principales activités : l'une centrée sur le développement de systèmes électroniques s'inspirant du système nerveux, l'autre s'intéressant davantage à l'élaboration d'outils à visées applicatives dans le domaine biomédical.

La richesse de chacune de ces thématiques a donné naissance au 1<sup>er</sup> Janvier 2010 à deux

nouvelles équipes, AS2N (Architecture of Silicon Neural Networks) et Elibio (Électronique pour l'Interaction avec la Biologie). Cependant, elles maintiennent toutes deux de fortes interactions, notamment du fait de l'utilisation d'outils mathématiques communs. Les travaux exposés dans ce manuscrit ont été poursuivis dans la première, celle-ci s'attachant au développement de circuits neuromimétiques, à la modélisation de neurones biologiques et à l'étude théorique de leurs comportements. Néanmoins, la scission étant récente, nous décrivons ci-dessous les projets pluridisciplinaires initiés dans l'équipe ISN, la nouvelle équipe étant trop jeune pour faire état d'une rétrospective.

### Les projets d'interface et d'instrumentation

Une partie des activités de l'équipe ISN reprise aujourd'hui par Elibio était dédiée au développement d'une instrumentation pour la réalisation d'interfaces vivant/artificiel. Les projets présentés ci-dessous, issus de collaborations tant régionales que nationales, européennes ou internationales, s'inscrivent dans cette problématique.

- Neuro-vers-IT (2006–2009) : il s'agissait d'un programme européen de formation à la recherche et de mobilité de type Marie-Curie (Research Training Network) entre équipes de recherche en neurosciences, sciences cognitives, robotique, micro-électronique et biomédical. L'objectif était ici de former de jeunes chercheurs (doctorants et post-doctorants) à l'ingénierie et la robotique bio-inspirées. (Soutien Région Aquitaine et Ferder).
- Delivrer (2007–2010) : cette collaboration régionale regroupe des laboratoires de recherche bordelais en chimie, biologie, bio- et micro-électronique et médecine. Elle vise à l'élaboration d'un système de contrôle en "boucle fermée" de la libération d'insuline chez les patients diabétiques. La régulation du taux d'insuline se fait à partir de l'enregistrement et de l'analyse de l'activité de cellules bêta-pancréatiques cultivées in-vitro. Là encore, l'équipe a pour rôle le développement logiciel et matériel de l'instrumentation en boucle fermée.
- STN oscillations (ANR, 2009–2012) : le dernier projet de la thématique concerne le traitement symptomatique des tremblements liés à la maladie de Parkinson. Ce projet bordelais a pour premier objectif de vérifier la validité de l'hypothèse selon laquelle une activité du noyau subthalamique dans une certaine gamme de fréquences serait à l'origine des tremblements, le second objectif est thérapeutique. L'équipe cherche à développer l'instrumentation nécessaire aux expériences, ainsi qu'à concevoir un dispositif intégré (sur silicium) afin de préparer la production de systèmes implantables et autonomes.

### Les projets neuromimétiques

La contribution de l'équipe nouvellement AS2N à ces projets est la conception de circuits neuromimétiques afin de mieux comprendre le fonctionnement du système nerveux ou à des fins computationnelles.

- FACETS et FACETS-ITN (2005–2014) : ce consortium européen réunit une douzaine d'équipes européennes. Le but de ce projet est de réaliser un système s'inspirant du fonctionnement cérébral afin d'élaborer de nouveaux paradigmes de calculs et de mieux appréhender le fonctionnement du système nerveux. Les groupes de recherche impliqués travaillent à différentes échelles de la modélisation du système (échelle cellulaire, réseau de neurones, système visuel, etc) et avec différents outils (expérimentation biologique, implémentation logicielle, implantation matérielle à l'aide de circuits analogiques, etc). L'équipe traite les niveaux de la cellule et du petit réseau de neurones ; elle développe des circuits

---

intégrés analogiques neuromimétiques fonctionnant en temps réel biologique.

- PIR CNRS NeuroInf ECRéN (2009–2011) : ce projet français issu d’une collaboration avec l’UNIC a pour objet l’étude des “fonctions de transfert” des réseaux de neurones par des méthodes combinatoires et en utilisant des circuits intégrés neuromimétiques. Il s’agit tout d’abord de créer une grande base de données des comportements d’un réseau pour ensuite les trier et les analyser afin d’en extraire des propriétés relatives à la structure et à la nature du réseau (connectivité, propriétés intrinsèques des cellules, des synapses,...). La nouvelle base de données ainsi créée sera alors mise à disposition de la communauté scientifique.

Ces travaux de thèse s’inscrivent en partie dans le cadre du projet européen FACETS. L’extraction de modèles de neurones biologiques est à un enjeu pour la communauté des neurosciences computationnelles : dans le but de reproduire un comportement dynamique, il est nécessaire d’isoler un modèle autorisant la reproduction d’activités variées, mais aussi de trouver les paramètres à appliquer à ce modèle afin qu’il exhibe un comportement particulier. C’est également un enjeu au sein de l’équipe car les circuits neuromimétiques sans réglage préalable ne peuvent fournir les comportements attendus en fonction du type de neurone dont on souhaite reproduire l’activité (neurone cortical, moto-neurone, etc). Enfin, les circuits une fois réglés constitueront eux aussi un outil d’étude du système nerveux au service des neurophysiologistes, aussi bien par le biais d’une interface hybride vivant/artificiel que par la mise en réseau de neurones artificiels complexes. Il est en effet quasiment impossible d’étudier expérimentalement avec précision le comportement électrique de petits réseaux de neurones vivants ; l’étude d’un réseau artificiel constitué de modèles de cellules complexes représente une alternative prometteuse.

Ce sujet avait pour principaux objectifs le développement d’une méthode permettant l’estimation des paramètres de modèles neuronaux, dans un premier temps, à l’échelle de la cellule, grâce à des enregistrements de courants ioniques – limitant ainsi le nombre de paramètres et la variété des signaux servant à l’estimation– puis à l’aide d’enregistrements de la tension de membrane d’un neurone – problématique plus complexe. Il s’agissait ensuite, si le temps le permettait, d’aborder la question de l’ajustement d’un petit réseau de neurones, ce qui requerrait, au préalable, une étude de la dynamique d’un réseau afin d’identifier les caractéristiques à reproduire, l’influence de la connectivité, l’influence des paramètres intrinsèques à une cellule sur le comportement du réseau, etc. Il était bien entendu impossible de réaliser cette dernière étude de manière exhaustive ; nous en avons posé les premières pierres. Le manuscrit de thèse est donc organisé en quatre chapitres comme suit :

1. Le premier fournit les éléments de bases de neurophysiologie nécessaires à la compréhension des modèles de neurones les plus usités que nous présentons dans ce même chapitre. Nous y introduisons également les éléments fondamentaux de l’optimisation, puis discutons du lien entre modélisation et optimisation afin de mieux cerner la problématique de nos travaux, en particulier l’estimation des paramètres de modèles de neurones.
2. Le deuxième chapitre s’intéresse à la recherche et au développement d’une méthode d’optimisation permettant d’estimer les paramètres du modèle complexe d’Hodgkin-Huxley à partir d’enregistrements de courants ioniques neuronaux.
3. Le troisième chapitre illustre l’utilisation de la méthode développée au chapitre précédent en l’appliquant à des mesures biologiques ainsi qu’à des données issues d’un circuit neuromimétique. Puis, dans une troisième section, nous étendons la technique d’optimisation à une extraction de paramètres utilisant uniquement des enregistrements de la tension de membrane d’un neurone.

4. Le chapitre 4 constitue quant à lui une ouverture vers le “réglage” d’un réseau de neurones ; nous nous focalisons ici sur la reproduction d’un certain type d’activité, caractéristique notamment du fonctionnement de l’hippocampe et du cortex auditif et considéré comme l’un des vecteurs de l’information neuronale dans ces structures cérébrales : les oscillations gamma. Dans ce contexte, nous étudions l’influence des propriétés intrinsèques des neurones sur l’activité globale d’un petit réseau d’une centaine de neurones.

Nous terminons finalement ce manuscrit par une conclusion dans laquelle sont présentées les perspectives de ces travaux à court et moyen terme.

# Chapitre 1

## Du neurone biologique au modèle de neurone

### Sommaire

---

<b>1.1</b>	<b>La modélisation de neurones biologiques : de l'anatomie au modèle</b>	<b>6</b>
1.1.1	Anatomie et physiologie d'un neurone biologique . . . . .	6
1.1.2	Les modèles mathématiques de la littérature . . . . .	10
<b>1.2</b>	<b>Les méthodes d'optimisation appliquées à l'estimation des paramètres de modèles de neurones : état de l'art</b> . . . . .	<b>14</b>
1.2.1	Quelques principes généraux en optimisation pour l'estimation de paramètres de modèles de neurones . . . . .	15
1.2.2	Application à l'estimation des paramètres de modèles de neurones . . . . .	16
<b>1.3</b>	<b>Retour sur le formalisme d'Hodgkin–Huxley et la méthode du voltage-clamp</b> . . . . .	<b>23</b>
1.3.1	Le formalisme d'Hodgkin–Huxley : une analogie biologie/électricité . . . . .	23
1.3.2	Le voltage-clamp et la méthode disjointe : limitations . . . . .	27
<b>1.4</b>	<b>Résumé</b> . . . . .	<b>35</b>

---

« οἶδα ὅτι οὐδὲν οἶδα »

(Attribué à Socrate par Platon)

## Introduction

Ce chapitre a pour but de présenter les notions de base nécessaires à la modélisation de neurones biologiques. La construction d'un paradigme est à la fois contrainte par l'observation et par ce que l'on souhaite reproduire : dans quel but ? À quelle échelle ? Quel comportement ? Le chapitre explique en même temps la nécessité d'ajuster les modèles par des méthodes d'optimisation. Pour éclairer ce cheminement, du neurone vers son modèle, la première partie s'organise suivant deux sections : la première vise à introduire les notions simples et fondamentales de l'anatomie et de la physiologie du neurone pour en venir, dans la seconde partie, aux modèles existants de la littérature, du modèle Integrate-and-Fire aux modèles à conductances. La deuxième partie met en évidence la relation étroite entre modélisation et optimisation en présentant tout d'abord les principes généraux de l'optimisation, puis en se recentrant sur l'utilisation des techniques d'optimisation dans la modélisation des neurones à l'échelle de la cellule avec les modèles à conductances. La fin du chapitre se focalise finalement sur le modèle et les méthodes qui ont retenu notre attention dans l'élaboration de la première partie de ce travail de thèse, à savoir le formalisme d'Hodgkin–Huxley ainsi que la méthode disjointe d'estimation associée au voltage-clamp et ses limitations.

### 1.1 La modélisation de neurones biologiques : de l'anatomie au modèle

#### 1.1.1 Anatomie et physiologie d'un neurone biologique

##### L'anatomie du neurone

Il existe une grande diversité de neurones (figure 1.1). Ils ont été classés selon leur morphologie qui reflète à la fois l'espèce animale, leur localisation dans le système nerveux et leur fonction. Il a cependant été dégagé une architecture globale, commune à tous les types de neurones chez lesquels on retrouve toujours dendrite(s), axone, synapses et corps cellulaire, appelé également soma (figure 1.2). Les différentes classes de neurones se distinguent essentiellement par la forme et la taille du soma et le nombre, la longueur et la répartition des dendrites. La description ci-après détaille le rôle de chacun de ces éléments.

- **Les dendrites**

Les dendrites sont des ramifications issues du corps cellulaire et qui permettent les connexions synaptiques afférentes au neurone. Elles assurent ainsi la propagation de l'information entre les différents neurones impliqués dans la communication en transmettant les informations générées au niveau synaptique préférentiellement vers le soma. Morphologiquement, elles se différencient de l'axone par des contours irréguliers (présence de ribosomes, petites protubérances synthétisant des protéines) et un diamètre diminuant au fur et à mesure qu'elles s'éloignent du soma. L'essentiel de l'information transmise d'un neurone à l'autre est reçue par les dendrites.

- **L'axone**

L'axone, généralement unique, est un prolongement long et cylindrique du soma. Son diamètre est inférieur à celui d'une dendrite et, chez l'homme, sa longueur peut être de l'ordre du mètre. Il conduit vers d'autres neurones un signal, le potentiel d'action, qui naît dans son segment initial. L'axone peut être entouré d'une gaine de myéline (substance essen-

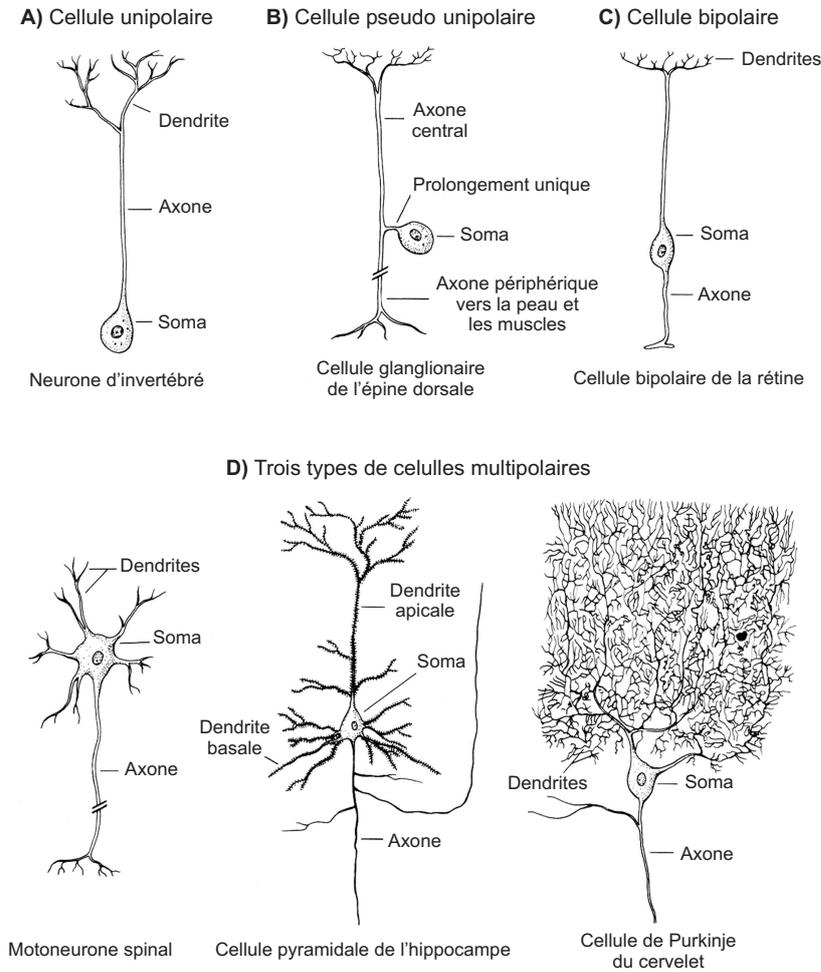


FIG. 1.1 – Illustration de la diversité morphologique des neurones [KSJ00].

tiellement constituée de lipides, isolant et protégeant l'axone) qui accélère la propagation du signal. On trouve, le long de l'axone, des points où la gaine de myéline s'interrompt ; ces points sont appelés noeuds de Ranvier. C'est au niveau de ces noeuds que le signal électrique se régénère.

#### – Les synapses

Du grec *συν* (« sun », avec, ensemble) et *ἅπτειν* (« haptain », toucher). Les synapses sont les points de contact entre un neurone et sa cible, un autre neurone, une fibre musculaire, etc... On les retrouve aussi bien sur le soma que sur l'arbre dendritique du neurone post-synaptique. En revanche, concernant le neurone pré-synaptique, elles se situent uniquement au niveau de ses terminaisons axonales.

On distingue dans le vivant deux catégories de synapses : les synapses dites « chimiques » et les synapses « électriques ». Les premières, rencontrées le plus fréquemment, nécessitent un intermédiaire pour transmettre l'information : le neurotransmetteur. En effet, l'activité électrique arrivant dans la terminaison axonale présynaptique stimule des vésicules synaptiques. Ces vésicules libèrent alors des neurotransmetteurs dans la fente synaptique

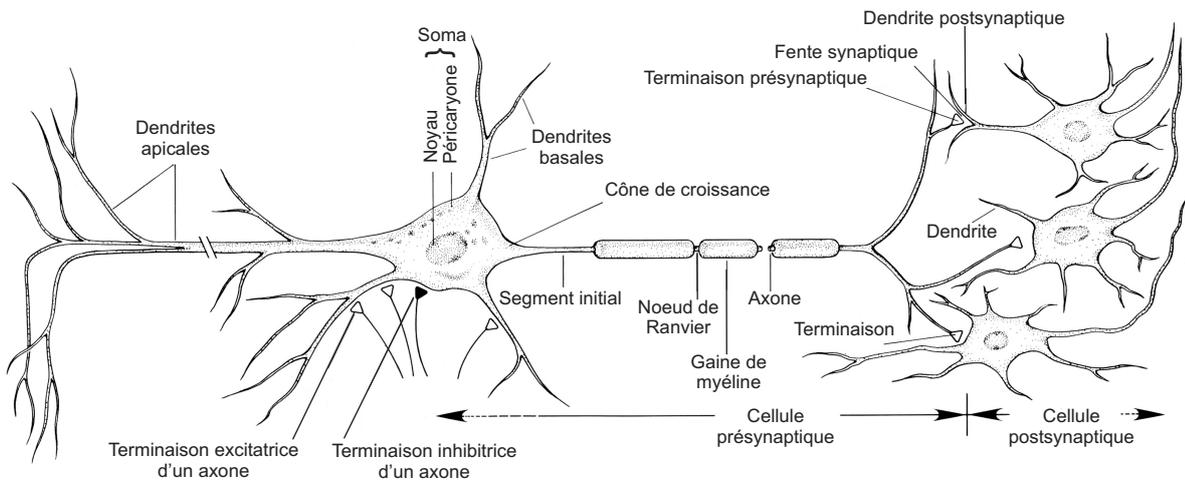


FIG. 1.2 – Schéma d'un neurone biologique [KSJ00].

et des récepteurs commandent l'ouverture de canaux ioniques particuliers induisant un courant post-synaptique dans les dendrites. Ces dernières transmettent ensuite à leur tour l'information au soma.

Dans le cas d'une synapse électrique, le signal est transmis directement au neurone post-synaptique sans délai de transmission.

On estime qu'environ 40 % de la surface membranaire de certains types de neurones (comme les cellules de Purkinje) sont couverts de synapses.

#### – Le soma et la membrane plasmique

Le soma (ou péricaryon) d'un neurone du cortex humain a un diamètre de l'ordre de  $20\mu m$  pour une épaisseur de membrane d'environ  $5nm$ . Les signaux arrivant sur les dendrites sont intégrés au niveau du soma qui génère à son tour une réponse se propageant dans l'axone.

La membrane plasmique d'une cellule en délimite le contour. Elle est la frontière entre les milieux intra- et extracellulaire et se compose d'une bicouche lipidique traversée par plusieurs types de protéines. Les milieux intra- et extracellulaire possèdent les mêmes espèces ioniques en des concentrations différentes de part et d'autre de la membrane. Le milieu extracellulaire est plus riche en ions sodium  $Na^+$  mais plus pauvre en ions potassium  $K^+$  que le milieu intracellulaire (figure 1.3). Il existe d'autres types d'ions, notamment calciques  $Ca^{2+}$ . Les différences de concentration calcique sont souvent proportionnelles aux variations de potentiel du neurone et peuvent être ainsi utilisées comme marqueur de l'activité électrique neuronale. Ce point sera plus clairement exposé dans la section suivante.

### La physiologie du neurone

Explicitons maintenant l'aspect fonctionnel des éléments décrits précédemment.

#### – Le potentiel de repos

Lorsque celle-ci n'est soumise à aucune excitation, le système constitué de la membrane et

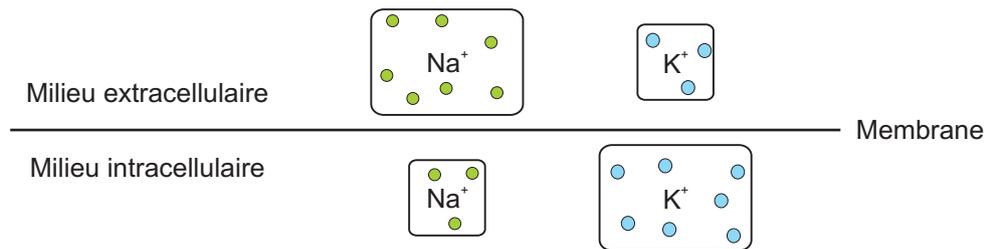


FIG. 1.3 – Schéma de la membrane plasmique d'un neurone.

des milieux extra- et intracellulaire est à l'équilibre, mais il existe malgré tout une différence de potentiel due à l'inégalité de concentration des ions de part et d'autre de la membrane. Cette différence de potentiel prend toujours pour référence le milieu extracellulaire et porte le nom de potentiel de repos. Ce dernier est généralement de l'ordre de  $-65mV$ .

En effet, la bicouche lipidique possède deux catégories de protéines transmembranaires. La première permet le transfert spécifique d'une espèce ionique particulière. Ce transfert s'effectue grâce à la pression osmotique et au champ électrique transmembranaire. La seconde catégorie de protéines maintient une différence de concentrations ioniques entre les deux milieux : on parle alors de pompage actif. Celui-ci permet de conserver un gradient de concentration de part et d'autre de la membrane contrebalançant la diffusion osmotique des espèces ioniques au travers des canaux lorsqu'ils s'ouvrent. La membrane au repos est plus perméable aux ions  $K^+$  qu'aux autres ions [PAFH05] [HK49].

#### – Le potentiel d'action

Lorsqu'une stimulation est imposée au neurone, sa membrane devient extrêmement perméable aux ions  $Na^+$ . Les propriétés de perméabilité sélective des protéines transmembranaires donnent naissance à un signal électrique se propageant le long de l'axone : le potentiel d'action. La figure 1.4 illustre ce phénomène en cinq étapes.

1. Lors de la stimulation, la membrane commence par se dépolariiser.
2. Si la stimulation persiste, la membrane se dépolariise jusqu'à atteindre le seuil de déclenchement (du potentiel d'action) et les canaux perméables aux ions sodium s'ouvrent rapidement, accentuant encore la dépolariisation. A noter que le passage des ions sodium continue si le seuil a été franchi, même si la stimulation a cessé.
3. Lorsque la tension de membrane a atteint environ  $58mV$  [HK49], les canaux sodiques se ferment spontanément, ce qui a pour effet l'arrêt de l'augmentation de la dépolariisation.
4. Peu après, un second processus entre en action : l'ouverture des canaux potassiques. Ce phénomène provoque la repolarisation de la membrane plasmique. Cette période de décroissance du potentiel d'action est appelée « période réfractaire absolue » car un second potentiel d'action ne peut apparaître durant cette période du fait de l'inactivation des canaux sodiques.
5. Le délai de fermeture des canaux potassiques provoque ensuite une hyperpolarisation transitoire (ou période réfractaire relative) de la membrane qui retrouvera son potentiel de repos après la fermeture des canaux potassiques.

Après avoir été généré dans le corps cellulaire, ce potentiel d'action va être transmis à d'autres neurones via l'axone sans atténuation. En effet, une « régénération » du potentiel d'action a lieu

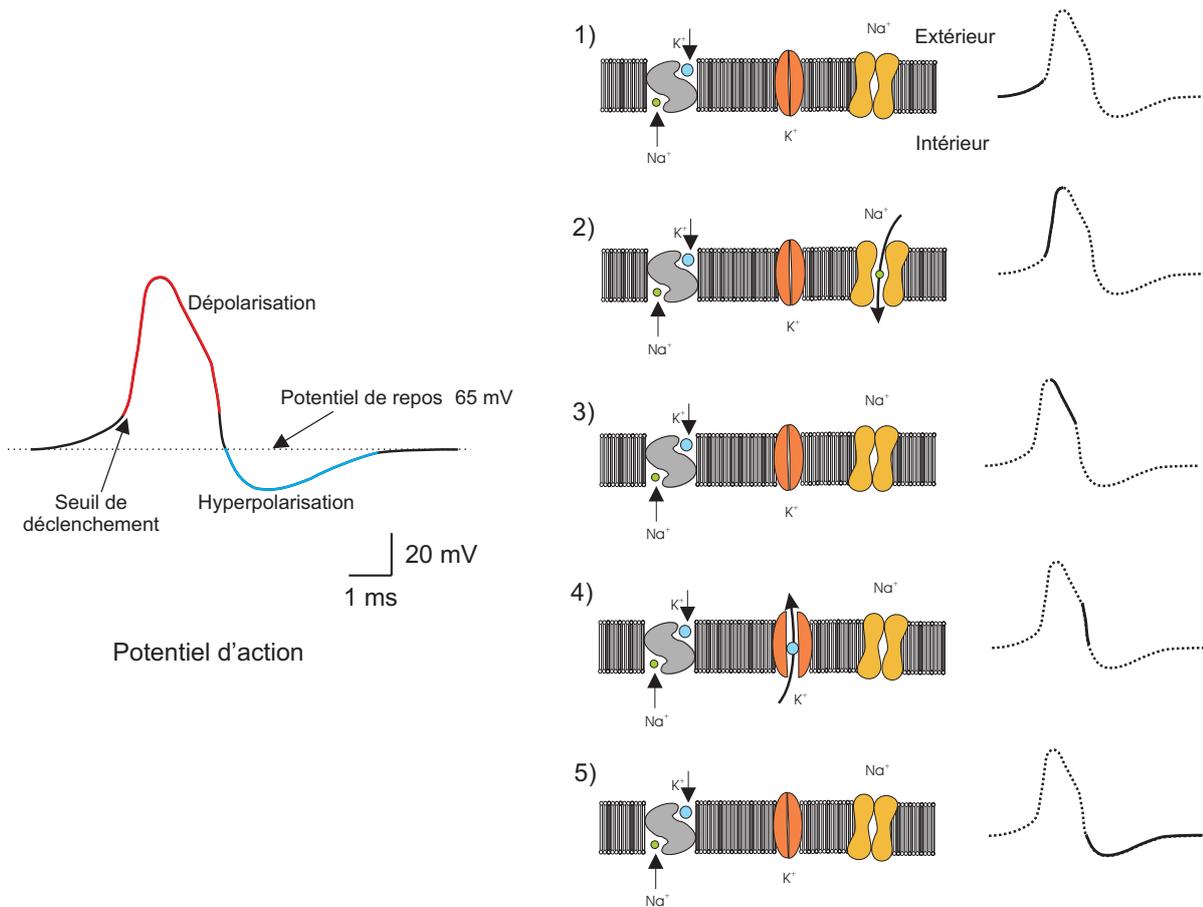


FIG. 1.4 – Mécanismes de genèse d'un potentiel d'action.

à chaque noeud de Ranvier jusqu'aux terminaisons axonales : le potentiel de l'axone au noeud suivant est entraîné au-dessus du seuil de déclenchement.

### 1.1.2 Les modèles mathématiques de la littérature

Un modèle mathématique de neurone biologique vise à décrire le neurone grâce à un formalisme mathématique afin de reproduire son activité. Les modèles actuels cherchent à imiter l'activité électrique et, bien que l'influence de processus thermodynamiques ait été démontrée ([LVRE02]), aucun modèle thermodynamique de l'activité neuronale n'est utilisé à ce jour. Selon la fonction du modèle et le type d'activité que l'on cherche à reproduire (activité d'un grand réseau de plusieurs milliers de neurones, d'un petit réseau de quelques centaines de cellules, d'un seul neurone et de ses éléments constitutifs, etc), plusieurs choix se présentent ; à noter que plus le modèle est complexe, plus les ressources calculatoires nécessaires à la réalisation d'une simulation logicielle sont importantes et plus le temps de calcul s'éloignera du temps réel biologique<sup>1</sup>. La complexification du formalisme entraînera également des problèmes d'ajustement du modèle avec l'augmentation du nombre de paramètres et la non linéarité des équations.

Dans cette section, nous présentons un état de l'art des modèles mathématiques de neurones

<sup>1</sup>Puisque les modèles utilisés reproduisent une activité électrique, un moyen de passer outre la complexité calculatoire et le temps de calcul est de concevoir des circuits électroniques neuromimétiques analogiques.

biologiques ou modèles impulsionnels en cherchant à faire ressortir à la fois les applications possibles et le degré d'ajustement des modèles. Ceux-ci sont présentés du modèle le plus élémentaire aux modèles les plus complexes. Le formalisme d'Hodgkin et Huxley présenté succinctement ici, fera l'objet d'un développement à la section 1.3 qui lui est consacrée.

### Les modèles Integrate-and-Fire (IF) et Leaky-Integrate-and-Fire (LIF)

- Le modèle « intègre et tire », en anglais « Integrate-and-Fire » ([Lap]), a été introduit au tout début du vingtième siècle par Louis Edouard Lapicque (1866-1952), médecin et physiologiste français, spécialiste du système nerveux. Sa représentation de l'activité électrique du neurone est issue d'un modèle de charge et décharge d'un condensateur à travers une résistance, dont l'équation figure ci-dessous :

$$I(t) = C_m \frac{dV_{\text{MEM}}}{dt} \quad (1.1)$$

où  $C_m$  représente la capacité de membrane,  $V_{\text{MEM}}$  la tension de membrane et  $I$ , le courant. Ce modèle ne rend pas compte de l'apparition d'un potentiel d'action, s'éloignant en ce sens d'un neurone vivant. Pour rendre compte de l'apparition d'un potentiel d'action, on introduit des distributions de Dirac,  $\delta(t)$ , qui sont des modèles des potentiels d'action. Une impulsion est délivrée chaque fois que la valeur du potentiel de membrane est égale à celle d'un seuil  $V_s$  (généralement  $-50mV$ ) et après chaque émission, le potentiel est ramené à sa valeur de repos ( $-65mV$ ). Pour plus de réalisme, il est possible d'ajouter à ce modèle une période réfractaire ; celle-ci permet de rendre compte du fait, observé expérimentalement, que la fréquence d'apparition des potentiels d'action est limitée même si le neurone est maintenu constamment dans des conditions où il devrait émettre un potentiel d'action.

- Un modèle « intègre et tire avec fuite » ou « Leaky-Integrate-and-Fire » ([Ste67]) a ensuite été proposé pour ramener progressivement le neurone à son potentiel de repos. L'équation 1.1 devient alors :

$$I(t) - \frac{V_{\text{MEM}}}{R_m} = C_m \frac{dV_{\text{MEM}}}{dt} \quad (1.2)$$

où  $R_m$  est la résistance membranaire. Ainsi, pour émettre un potentiel d'action, le courant entrant doit dépasser le seuil  $I_s = V_s/R_m$ . L'équation peut encore s'écrire :

$$C_m \frac{dV_{\text{MEM}}}{dt} = I(t) - g_{fuite}(V_{\text{MEM}} - E_{fuite}) \quad (1.3)$$

Ces deux variantes présentent peu de caractéristiques à ajuster, le seuil  $V_s$  et la fréquence d'oscillation, ce qui revient encore à régler la stimulation,  $g_{fuite}$  et  $E_{fuite}$ . Elles sont donc aujourd'hui utilisées pour simuler de très grands réseaux de neurones comme des structures cérébrales (colonnes corticales cortex visuel) où seule l'activité globale du système importe ([COS<sup>+</sup>02], [KH05], [RC07]). De tels modèles peuvent être réglés "manuellement".

### Les modèles Resonate-and-Fire, IF quadratique et d'Izhikevich

Des extensions du model IF, plus réalistes phénoménologiquement, ont été suggérées sur la base de l'équation 1.3.

- La première a été proposée en 1937 par Gaylord Young ([You37]) sous le nom de « résonne et tire » ou « Resonate-and-Fire » (RF) introduisant un courant potassique à faible seuil de déclenchement. Il est décrit par le système d'équations 1.5. Soit  $W$  le courant potassique supplémentaire et  $V_{\text{MEM}}$  la tension de membrane.

$$C_m \frac{dV_{\text{MEM}}(t)}{dt} = I(t) - g_{\text{fuite}}(V_{\text{MEM}}(t) - E_{\text{fuite}}) - W(t) \quad (1.4)$$

$$\tau \frac{dW(t)}{dt} = \frac{(V_{\text{MEM}}(t) - V_0)}{k} - W(t) \quad (1.5)$$

où  $\tau$ ,  $k$  et  $V_0$  sont des paramètres à fixer.  $V_{\text{MEM}}$  et  $W$  sont ramenés respectivement au potentiel de repos et à une valeur  $W_r$  si  $V_{\text{MEM}}$  dépasse le seuil  $V_s$ . Dans ce modèle,  $\tau$ ,  $V_0$ ,  $k$  et  $W_r$  sont des paramètres supplémentaires à ajuster.

- La seconde est appelée IF quadratique car elle fait intervenir le carré du potentiel de membrane :

$$C_m \frac{dV_{\text{MEM}}(t)}{dt} = I(t) + k_0 V_{\text{MEM}}(t)^2 \quad (1.6)$$

Comme pour ses prédécesseurs, ce modèle ramène le potentiel de membrane à une valeur d'équilibre si  $V_{\text{MEM}}$  dépasse un certain seuil. Le nombre de paramètres à estimer est le même que pour un modèle IF classique.

- La troisième, introduite par Eugene Izhikevich ([Izh03]), est une combinaison des deux précédentes ; elle fait à la fois intervenir l'expression quadratique et le courant potassique à faible seuil. Il est formalisé par les équations suivantes où  $v$  représente une tension,  $u$  un courant et  $a$ ,  $b$ ,  $c$ ,  $d$  sont des constantes :

$$\frac{dv(t)}{dt} = 140 + 0.04v(t)^2 + 5v(t) - u(t) \quad (1.7)$$

$$\frac{du(t)}{dt} = a(b.v(t) - u(t)) \quad (1.8)$$

$$\text{et si } v \geq 1 \text{ alors } v \leftarrow c, u \leftarrow u + d \quad (1.9)$$

Bien que relativement faciles à résoudre numériquement, les équations de ces modèles fournissent des comportements plus réalistes que les modèles IF classiques. Comme ces derniers, les modèles RF, IF quadratique et d'Izhikevich sont également utilisés pour modéliser de grands réseaux de neurones ou des structures cérébrales telles que celles impliquées dans la maladie de Parkinson comme dans [MMB07] pour le modèle d'Izhikevich. Les modèles de G. Young et d'E. Izhikevich ont davantage de paramètres à équilibrer selon le type de neurone à modéliser. L'utilité de techniques de réglage automatique se fait alors ressentir...

## Le formalisme d'Hodgkin–Huxley

Ce formalisme fait partie des formalismes dits « à conductances ». A la différence des modèles précédents qui étaient plus abstraits, le formalisme d'Hodgkin–Huxley<sup>2</sup> s'inscrit dans une modélisation biophysique à l'échelle des sous-éléments constituant la cellule. Les différents éléments constitutifs de la membrane sont assimilés à des grandeurs électriques (capacité, conductances,

<sup>2</sup>Dans la suite, nous appelons “formalisme d'Hodgkin–Huxley” tout jeu d'équations du type 1.8, quelque soit le nombre de courants ioniques modélisés.

...). Ce modèle décrit l'activité du neurone au niveau cellulaire incluant les échanges ioniques par les courants  $I_{\text{ion}}$  de l'équation ci-dessous .

$$C_{\text{MEM}} \frac{dV_{\text{MEM}}(t)}{dt} = \sum_{\text{ion}} I_{\text{ion}}(V_{\text{MEM}}(t)) \quad (1.10)$$

$$I_{\text{ion}}(t) = g_{\text{ion}}(V_{\text{MEM}}(t))(V_{\text{MEM}}(t) - E_{\text{ion}}) \quad (1.11)$$

où la conductance  $g_{\text{ion}}$  s'exprime ainsi :

$$g_{\text{ion}}(V_{\text{MEM}}(t)) = g_{\text{ion}}^- \cdot m(V_{\text{MEM}}(t))^p \cdot h(V_{\text{MEM}}(t))^q \cdot (V_{\text{MEM}}(t) - E_{\text{ion}}) \quad \text{avec} \quad (1.12)$$

$$\tau_x(V_{\text{MEM}}(t)) \frac{dx(t)}{dt} = x_{\infty} - x(t), \quad x = m, h \quad (1.13)$$

$\tau_x$  est une fonction homogène à un temps et qui dépend de 1 à 4 paramètres.

Initialement présenté par Hodgkin et Huxley avec deux courants ioniques (potassium et sodium), ce formalisme peut être étendu à d'autres classes de neurones comportant divers types de canaux ioniques (potassium dépendant du calcium, courants faible seuil,...) et est aujourd'hui utilisé pour modéliser des neurones à l'échelle cellulaire comme dans [PTRM<sup>+</sup>08] ou de petits réseaux d'une centaine de neurones. Dans le modèle simplifié à deux courants ioniques, il faut estimer une vingtaine de paramètres ; plus on ajoute de canaux ioniques, plus il y a de paramètres à ajuster. Le réglage empirique est donc très laborieux. Nous reviendrons en détail sur ce formalisme à la section 1.3 dans laquelle nous détaillerons les équations et la construction du modèle.

### Les réductions du modèle d'Hodgkin–Huxley : modèles de FitzHugh–Nagumo, Morris–Lecar et Hindmarsh–Rose

Du fait de sa complexité de réglage le formalisme d'Hodgkin a été décliné en variantes simplifiées par Fitz Hugh et Nagumo en 1961 ([Fit55], [Fit61]) et 1962 ([NAY62]), Morris et Lecar en 1981 ([ML81]), Hindmarsh et Rose en 1984 ([HR84]) ou Doya en 1993 ([DSR93]). Elles portent maintenant leur nom.

Nous présentons brièvement les équations de quelques unes d'entre elles :

- Equations du modèle de FitzHugh–Nagumo :

$$\frac{dV_{\text{MEM}}(t)}{dt} = V_{\text{MEM}}(t) - V_{\text{MEM}}^3(t) - w(t) - I(t) \quad (1.14)$$

$$\frac{dw(t)}{dt} = V_{\text{MEM}}(t) - a - bw(t) \quad (1.15)$$

avec  $w$ , variable homogène à une tension,  $I$  un courant de stimulation extérieur,  $a$  et  $b$  des constantes à déterminer. (Notons qu'il ne faut pas oublier, pour l'homogénéité, de multiplier  $V_{\text{MEM}}$  par les unités appropriées dans le membre de droite des équations ci-dessus).

Ce modèle qui a donc deux paramètres à ajuster est employé dans la modélisation de très grands réseaux de neurones ([CM04]).

- Le modèle de Morris et Lecar est un modèle à conductances qui combine le formalisme original d'Hodgkin et Huxley et le modèle de FitzHugh–Nagumo ; il a plus de paramètres à ajuster que FitzHugh–Nagumo, mais moins qu'Hodgkin–Huxley (une dizaine). Il est préféré

pour la modélisation d'une cellule ou de réseaux de taille moyenne avec des comportements dynamiques variés. Il se présente sous la forme suivante :

$$C_{\text{MEM}} \frac{dV_{\text{MEM}}}{dt} = \sum_{\text{ion}} I_{\text{ion}}(V_{\text{MEM}}) \quad (1.16)$$

$$\frac{dw}{dt} = \phi \frac{w_{\infty} - w}{\tau_w(V_{\text{MEM}})} \quad (1.17)$$

$$\begin{aligned} \sum_{\text{ion}} I_{\text{ion}} &= g_{\text{Ca}}^{-} m_{\infty}(V_{\text{MEM}} - V_{\text{Ca}}) + g_{\text{K}}^{-} w \cdot (V_{\text{MEM}} - E_{\text{K}}) \\ &\quad + g_{\text{fuite}}^{-} \cdot (V_{\text{MEM}} - E_{\text{fuite}}) \end{aligned} \quad (1.18)$$

- Enfin, le modèle Hindmarsh-Rose, quant à lui, est construit à partir du modèle de FitzHugh–Nagumo et combine trois équations différentielles du premier ordre :

$$\frac{dx}{dt} = y + 3x^2 - x^3 - z + I \quad (1.19)$$

$$\frac{dy}{dt} = 1 - 5x^2 - y \quad (1.20)$$

$$\frac{dz}{dt} = r \cdot [4 \cdot (x + 8/5) - z] \quad (1.21)$$

Le paramètre à estimer est ici la constante  $r$ . Ce modèle simple est capable de reproduire des comportements dynamiques variés sans nécessiter beaucoup de réglages ; il est donc utilisé pour modéliser des réseaux à grande échelle ([XT08]), à l'instar des modèles IF, d'Izhikevich ou FitzHugh–Nagumo.

## Les modèles multi-compartimentaux

Le formalisme d'Hodgkin–Huxley ou les autres modèles à conductances sont également souvent utilisés pour la conception de modèles de neurones comportant plusieurs *compartiments*. On modélise alors axones, dendrites, synapses, épines dendritiques, etc, généralement sous la forme de cylindres ou d'un assemblage de cylindres. Les modèles à conductances sont alors complétés par la théorie des câbles pour introduire des effets dissipatifs et complexifier la dynamique. Ceci permet d'inclure de manière réaliste la géométrie du neurone et les conséquences fonctionnelles de la structure : les délais de propagation dans les différents compartiments, la communication entre neurones, des phénomènes de plasticité synaptique, etc. Les synapses, comme les canaux ioniques, peuvent être représentées par une conductance maximale et un potentiel d'équilibre combinés dans une (ou plusieurs) fonction(s) exponentielle(s) comme c'est le cas pour les modèles dits « synapse exponentielle », « synapse alpha », ou « synapse biexponentielle ». Dans de telles configurations, des centaines de paramètres peuvent être réglés.

Les modèles complexes tels qu'Hodgkin–Huxley ou les modèles multi-compartimentaux sont donc difficiles à ajuster manuellement. Des méthodes d'ajustement automatiques à l'aide de techniques d'optimisation peuvent alors apporter une aide précieuse.

## 1.2 Les méthodes d'optimisation appliquées à l'estimation des paramètres de modèles de neurones : état de l'art

Le but de cette section est de mettre en évidence le lien entre l'optimisation et la modélisation en neurosciences computationnelles. L'usage de techniques d'optimisation s'explique en

effet par la nécessité d'ajuster les modèles pour les adapter aux comportements que l'on souhaite reproduire (comportement global d'un réseau de neurones, échelle mésoscopique, échelle microscopique...). En guise de préliminaire, nous présentons ce en quoi consiste l'optimisation : nous en donnons les principes fondamentaux, puis décrivons les méthodes les plus utilisées pour modéliser les neurones avant de dresser un état de l'art concernant l'optimisation pour la modélisation neuronale.

### 1.2.1 Quelques principes généraux en optimisation pour l'estimation de paramètres de modèles de neurones

L'optimisation vise à minimiser ou maximiser une fonction  $f$  définie sur un ensemble  $A$  –  $A$  étant déterminé par la nature du problème – et à valeurs dans un espace  $B^n$ , continu ou discret. Elle consiste donc à rechercher l(es) élément(s)  $x^*$  tel(s) que :

$$\forall x \in A, f(x^*) \geq f(x) \text{ (maximisation)}$$

$$\forall x \in A, f(x^*) \leq f(x) \text{ (minimisation)}$$

Dans la suite, nous chercherons à minimiser<sup>3</sup> la fonction  $f$ . Cette fonction est usuellement appelée *fonction objectif*, *fonction coût* ou encore *fonction erreur*.

L'optimisation numérique, quant à elle, est généralement utilisée dans les cas où l'on ne sait pas résoudre analytiquement le problème de recherche de  $x^*$  : il s'agit d'approcher, par itérations successives, la valeur de  $x^*$ . C'est à l'optimisation numérique que nous nous intéresserons dans la suite de ce manuscrit.

Après avoir choisi le signal que nous souhaitons reproduire et qui nous servira de référence, trois choix sont prépondérants dans la résolution d'un problème d'optimisation dans le cadre de la modélisation de neurones biologiques :

1. Tout d'abord, celui du modèle permettant de reproduire un comportement visé. Autrement dit, il faut en premier lieu déterminer un jeu d'équations dont nous optimiserons les paramètres. Dans le cadre de la modélisation d'un neurone, il s'agit généralement de reproduire les oscillations de la tension de membrane ou un courant ionique.
2. Ensuite, s'impose le choix d'une fonction coût  $F_{cout}$  selon les caractéristiques du comportement visé. Ce choix est très important car il influencera la qualité de la solution et le choix de l'algorithme d'optimisation en fonction de la nature analytique de  $F_{cout}$  (minima locaux/globaux, discontinuités, etc...).
3. Enfin, la résolution du problème nécessite de choisir une méthode d'optimisation qui assure à la fois la convergence et l'obtention d'une solution « acceptable » en un temps « raisonnable ».

L'acceptabilité de la solution est fixée en fonction de la qualité attendue du résultat, soit par une contrainte sur la solution (variation des solutions au cours des dernières itérations inférieure à un certain seuil), soit, le plus fréquemment, par une contrainte sur la valeur de la fonction de coût (par exemple  $F_{cout}(x_{optimal})/\text{signal de référence} \leq \text{seuil}$ ). La notion de temps de résolution, quant à elle, dépend des contraintes extérieures : optimisation en temps réel, besoins industriels, moyens de calculs mis à disposition, etc...

---

<sup>3</sup>On remarquera que maximiser  $f$  revient à minimiser  $-f$

Parmi les techniques d'optimisation, on distingue, sommairement, les méthodes dites déterministes, comme celles de type *descente de gradient* et les méthodes stochastiques, telles que les *métaheuristiques*. Les méthodes de type descente de gradient consistent, à partir d'un point aléatoire de l'espace de départ  $A$ , à se déplacer dans la direction de la plus forte pente de  $f$ . En appliquant un certain nombre d'itérations, l'algorithme converge vers une solution qui est souvent un minimum local de  $f$ . Une façon d' "éviter" les minima locaux avec ce type de méthodes est de répéter l'algorithme avec une initialisation aléatoire différente à chaque essai. Les méthodes stochastiques sont quant à elles plus robustes vis à vis des minima locaux puisqu'elles font appel, dans leur propre réalisation, à des tirages aléatoires ; ce qui permet d'explorer efficacement l'espace de recherche. Parmi elles, les métaheuristiques<sup>4</sup> désignent des méthodes d'optimisation inspirées de phénomènes naturels, physiques ou biologiques. On classe souvent les métaheuristiques suivant deux types : le premier contenant les algorithmes à base de solution courante unique auquel appartient par exemple l'algorithme de *recuit simulé* s'inspirant de la méthode du recuit en métallurgie ; le second réunissant les techniques dites « à population » comme les *algorithmes évolutionnaires* prenant modèle sur les phénomènes de croisements et mutations observés dans le vivant avec l'évolution darwinienne des espèces. Pour des exemples d'algorithmes évolutionnaires, on pourra se reporter à l'article [DS08] qui fait une revue de quelques métaheuristiques à population. Dans le premier cas, un seul point de l'espace  $A$  est manipulé à chaque itération ; dans le second, on manipule un ensemble de points à chaque itération. On peut noter l'emploi d'un vocabulaire dédié pour les algorithmes évolutionnaires : le tableau 1.1 regroupe les terminologies usuellement adoptées en optimisation et en calcul évolutionnaire. Nous emploierons par la suite indifféremment les terminologies utilisées en optimisation et en optimisation évolutionnaire.

Mathématiques	Calcul évolutionnaire	Optimisation numérique
ensemble $A \subset \mathbb{R}^N \times \mathbb{R}^D$	population de $N$ individus	ensemble de solutions candidates
$D$ -vecteurs de paramètres	individus, descendance, parents	solutions candidates, variables de décision variables objet
un élément du $D$ -vecteur	un gène	un paramètre
fonction $f : A \rightarrow \mathbb{R}$ à minimiser	fonction d'adaptation (fitness)	fonction objectif, fonction de coût fonction erreur
itération	génération	itération

TAB. 1.1 – Terminologie des algorithmes évolutionnaires.

### 1.2.2 Application à l'estimation des paramètres de modèles de neurones

Nous nous focalisons ici sur les techniques employées pour l'estimation des paramètres de modèles de cellules, mono- ou multi-compartmentaux ; la question de l'ajustement d'un réseau de neurones sera abordée au dernier chapitre.

Les méthodes les plus utilisées jusqu'ici pour l'estimation de paramètres de modèles de neurones vivants ont été les méthodes de gradient, le recuit simulé et les algorithmes génétiques, exemple le plus connu d'algorithmes évolutionnaires. Décrivons donc préalablement ces méthodes dans des pseudo-codes avant de détailler leur emploi dans la littérature relativement aux modèles de neurones utilisés.

Dans la suite,  $x$  désigne un vecteur de paramètres de taille  $D$  s'il y a  $D$  paramètres à estimer,  $x_{opt}$  est le vecteur de paramètres optimal (au sens numérique et non analytique). Pour

<sup>4</sup>Le terme d'heuristique désigne en optimisation une méthode approximative, adaptée à un problème particulier, permettant d'en trouver une solution approchée en accélérant la recherche.

les algorithmes évolutionnaires,  $X$  représente une matrice de taille  $N_{ind} \times D$  où  $N_{ind}$  est le nombre d'individus de la population.

### Prélude algorithmique

---

**Algorithme 1** : Algorithme de descente de gradient simple

---

**Entrées** :  $N_{iter}$  : un nombre d'itérations,  $f$  : fonction coût

**Sorties** :  $x_{opt}$ , minimisant localement  $f$

**Initialisation** :  $x$  aléatoire tel que  $f$  différentiable en  $x$ ,  $\varepsilon$

**pour**  $k$  de 0 à  $N_{iter}$  **faire**

$x = x - \varepsilon \cdot \nabla f(x)$   
 $x_{opt} = x$

retourner  $x_{opt}$

---

Comme nous l'avons évoqué précédemment, la méthode de descente de gradient<sup>5</sup> simple (Algorithme 1) permet donc de rechercher des minima locaux. L'issue de l'optimisation dans le cas d'un problème comportant plusieurs minima dépend beaucoup de l'initialisation de l'algorithme dans la mesure où l'on s'oriente toujours dans le sens de la plus forte pente sans qu'aucune heuristique<sup>6</sup> ne permette de s'en "échapper" pour trouver un minimum global. La figure 1.5, illustre comment, à partir d'un point  $x_0$  aléatoire, on trouve le minimum local le plus proche. Sur la base de la descente de gradient, on retrouve également l'algorithme de gradient conjugué. Enfin, ces méthodes exigent de pouvoir approcher la dérivée de la fonction de coût, ce qui nécessite la dérivabilité de  $f$  et que les variables soient à valeurs continues. La descente de gradient ne peut donc pas s'appliquer à tous les problèmes de modélisation neuronale, ceux-ci mettant généralement en jeu des fonctions complexes dont on ne connaît pas nécessairement le comportement analytique.

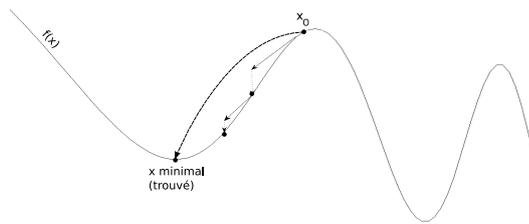


FIG. 1.5 – Illustration d'une "descente de gradient".

Le recuit simulé (Algorithme 2) quant à lui fait évoluer la solution en autorisant, à chaque itération et selon une probabilité qui diminue au cours du temps – et qui dépend de la valeur et

---

<sup>5</sup> $\nabla f(x) = f'(x)$  est le gradient de  $f$  en  $x$ . Lorsqu'on ne connaît pas l'expression analytique du gradient, sa valeur en  $x$  peut être approchée par discrétisation.

<sup>6</sup>Une heuristique désigne en mathématiques une méthode approximative permettant d'améliorer un algorithme ou d'accélérer le temps de résolution d'un problème.

de signe de la variation de la fonction de coût  $-$ , le remplacement de la solution courante  $x$  par une solution  $y$  de moins bonne qualité : ie telle que  $f(y) \geq f(x)$ . Cette technique permet ainsi de “s’échapper” d’un minimum local (figure 1.6) .

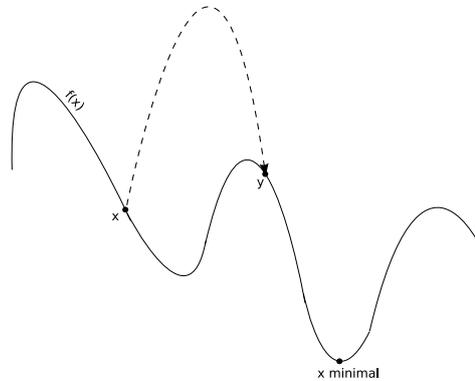


FIG. 1.6 – Illustration du “saut” hors d’un minimum local dans la méthode du recuit simulé.

---

### Algorithme 2 : Algorithme de recuit simulé

---

**Entrées** :  $f$  : fonction coût,  $T_{init}$  : température initiale,  $T_{fin}$  : température finale,  $N_{iter}$  : nombre d’itérations par pallier de température,  $h$  : fonction de la température.

**Sorties** :  $x_{opt}$ , minimisant  $f$ .

**Initialisation** :  $x$  aléatoire,  $x_{opt} = x$ ,  $f_{opt} = f(x_{opt})$ ,  $T = T_{init}$ ,  $k = 0$

```

tant que  $T > T_{fin}$  faire
  tant que  $k < N_{iter}$  faire
    Choisir un  $y$  dans le voisinage de  $x$ 
    calculer  $\Delta f = f(y) - f(x)$ 
    si  $\Delta f < 0$  alors
       $x = y$ 
      si  $f(x) < f(x_{opt})$  alors
         $x_{opt} = x$ 
         $f_{opt} = f(x_{opt})$ 
      sinon
        Tirer  $p$  dans  $[0,1]$ 
        si  $p \leq \exp(-\Delta f/T)$  alors
           $x = y$ 
     $k = k + 1$ 
   $T = h(T)$ 
retourner  $x_{opt}$ 

```

---

Les algorithmes génétiques (Algorithme 3) font évoluer à chaque itération (ou génération) un jeu de solutions, ou population d’individus, qui sont “recombinés” par échange de certains de leurs gènes et “mutés” par modification légère (seuil de modification à fixer en fonction de l’ordre de grandeur des paramètres en question) de leurs gènes. Ces deux opérations sont respectivement appelées *croisement* et *mutation*. Les algorithmes génétiques présentant différentes variantes, nous ne proposons ici que le principe général de ce type d’optimisation. Le schéma 1.7 en

illustre le cheminement.

---

**Algorithme 3** : Principe des algorithmes génétiques

---

**Entrées** :  $N_{iter}$  : un nombre d'itérations,  $f$  : fonction coût,  $pc$  : probabilité de croisement,  $pm$  : probabilité de mutation,  $Q_{opt}$  : proportion d'individus.

**Sorties** :  $x_{opt}$ , minimisant  $f$ .

**Initialisation** :  $X = X_0$  aléatoire,  $k = 0$ ,  $x_{opt}$

**pour**  $k$  de 0 à  $(N_{iter} - 1)$  **faire**

    Sélectionner et conserver une proportion  $Q_{opt}$  des meilleurs individus.

    Effectuer croisements entre ces individus par échange d'un ou plusieurs de leurs gènes avec une probabilité  $pc$

    Effectuer mutations par modification aléatoire d'un ou plusieurs gènes des nouveaux individus (enfants) avec une probabilité  $pm$

    Evaluer les enfants

    Créer  $X_{k+1}$  : remplacer des individus de  $X_k$  par les nouveaux individus (enfants) et/ou conserver les meilleurs individus de  $X_k$ .

retourner  $x_{opt}$

---

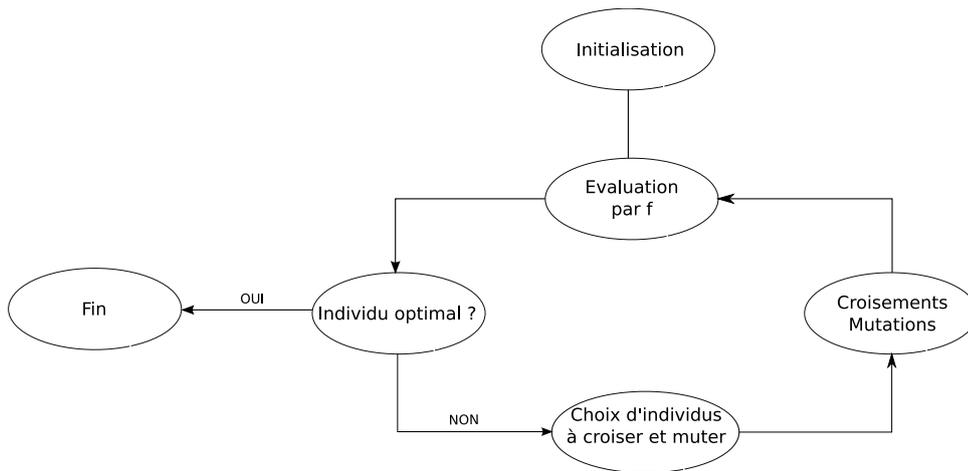


FIG. 1.7 – Algorithmes génétiques – Principe

Revenons-en maintenant à la modélisation neuronale pour nous intéresser aux modèles à conductance. Comme dans beaucoup de problèmes d'estimation de paramètres en biologie, les données enregistrées qui servent de référence pour la construction d'un modèle, sont parasitées par le bruit environnant (afférences synaptiques des neurones voisins, bruit électrique des appareils de mesure, etc...) ou soumises à des perturbation paramétriques (température de la pièce, pression, etc...). Le modèle, une fois l'optimisation effectuée, ne pourra donc pas reproduire à l'identique –c'est-à-dire avec une valeur nulle de la fonction de coût– le signal de référence. La difficulté réside donc dans l'adaptation de la méthode d'exploration au type d'enregistrements qui nous sont fournis, afin de s'approcher le plus possible du minimum global. Parmi les publications relatives aux méthodes automatisées d'estimation de paramètres de modèles de neurones, on peut majoritairement distinguer deux classes de travaux : ceux dans lesquels la fonction de

coût est construite à partir des courants ioniques, données dont l’obtention est contraignante pour les neurophysiologistes, et ceux se contentant d’enregistrements de la tension de membrane du neurone, signaux dont l’acquisition est plus aisée mais pour lesquels il y a davantage de paramètres à estimer.

### Estimation de paramètres à partir d’enregistrements de courants ioniques

L’optimisation à partir de courants ioniques consiste à comparer des courants ioniques simulés à des courants ioniques enregistrés. La plus célèbre des méthodes d’estimation des paramètres des canaux ioniques est la méthode disjointe associée au voltage-clamp et proposée par Hodgkin et Huxley en 1952. Cette méthode souffre malheureusement d’erreurs d’estimation importantes dues à des approximations. Nous y reviendrons en détail à la dernière section de ce chapitre (section 1.3). Plusieurs alternatives ont donc été proposées par la suite.

La première fut proposée en 1999 par Willms [WBHWG99] qui utilise une méthode plus “classique” : optimisation non-linéaire sous contraintes à l’aide de méthodes de type descente de gradient (Levenberg-Marquardt) pour minimiser l’erreur quadratique entre les courants calculés et de référence. Cependant, il reste difficile d’estimer les constantes de temps avec cette technique.

On retrouve aussi, parmi les alternatives, des travaux cherchant à exploiter les propriétés de parallélisation des algorithmes [CSHF08], [GK07] et des études cherchant à réduire l’espace de recherche [MMCS00][DOK02]. Dans toutes les optimisations réalisées, la fonction de coût est une erreur quadratique entre le courant estimé  $I_{est}$  et le courant de référence  $I_{ref}$  pour plusieurs valeurs de stimulations  $stim$  différentes sommées au cours du temps  $t$  :

$$F_{cout} = \sum_t \sum_{stim} (I_{est}(stim, t) - I_{ref}(stim, t))^2 \quad (1.22)$$

Estimer les paramètres d’un modèle en utilisant des informations sur les courants ioniques permet de subdiviser l’optimisation en sous-problèmes. Il en résulte généralement un critère par type d’ion, chaque critère pouvant être aggloméré aux autres par sommation dans une fonction de coût unique comme le font Dávid Csercsik et al. dans [CSHF08]. Les auteurs cherchent dans cet article à construire un modèle de neurone à GnRH (neurone de la région préoptico-hypothalamique des mammifères) à partir d’un formalisme d’Hodgkin–Huxley en estimant les quatre conductances maximales et la capacité de membrane du neurone, les autres paramètres étant supposés connus. La fonction de coût nécessitant un grand nombre de calculs, ils ont opté pour un algorithme de recherche parallèle, le Parallel Pattern Search (PPS) adapté aux problèmes caractérisés par un petit nombre de variables (inférieur à 100). Cet algorithme consiste à faire évoluer la solution courante  $x_k$  dans une certaine direction  $\Delta_k$  qui diminue au cours des itérations si  $x_k$  est meilleure que la solution optimale courante. Gurkiewicz et al. ont également proposé une méthode d’estimation tirant partie de la parallélisation : ils utilisent en effet un algorithme génétique implémenté sur 4 ordinateurs de type Pentium (3GHz). La convergence de leur estimation dépend malgré tout de l’initialisation et nécessite parfois d’être complétée par une recherche locale après avoir identifié la zone optimale.

La majorité des méthodes proposées demande donc beaucoup de ressources calculatoires et lorsque ce n’est pas le cas, la technique d’optimisation employée est potentiellement sujette à “tomber” dans des minima locaux.

### Estimation de paramètres à partir d’enregistrements de la tension de membrane

D’autres auteurs ont proposé d’estimer les paramètres de leurs modèles en utilisant directement des enregistrements de la tension de membrane. La plupart du temps, les modèles qu’ils

manipulent sont multi-compartimentaux et la technique neurophysiologique habituelle d'enregistrement des différents courants ioniques s'avère impossible sur tous les compartiments du neurone. Les paramètres estimés sont alors le plus souvent les conductances maximales, les autres paramètres étant supposés connus pour faciliter la recherche ([GSA07]). L'estimation de tous les paramètres de ces modèles s'avèrerait très coûteuse en temps et en calculs car jusqu'ici aucune méthode "économique" n'a encore été exhibée. Une revue des techniques d'optimisation utilisées pour l'estimation des paramètres de modèles de neurones est présentée dans [GSA08].

Les enregistrements de la tension de membrane d'un neurone présentent différentes caractéristiques qu'on peut ou non vouloir reproduire selon l'application visée comme nous l'avons déjà évoqué dans la partie 1.1.2 : la fréquence d'oscillation et sa variabilité, les phénomènes d'adaptation, l'amplitude du signal, la période, les valeurs minimale et maximale, l'évolution de la tension de membrane sous le seuil de déclenchement d'un potentiel d'action, etc. C'est le choix de ces critères qui est déterminant à la fois pour la forme du signal reproduit et dans le choix de la fonction coût. Par exemple, si l'on dispose d'un modèle du type IF et que seule la fréquence d'oscillation au-dessus du seuil importe, il sera judicieux de choisir une fonction de coût qui dépende de la fréquence comme la différence quadratique entre la fréquence de référence et la fréquence simulée d'émission du potentiel d'action. Notons par ailleurs que les mesures relatives aux critères qui nous intéressent doivent être adaptées au type de signal de référence (enregistrements physiologiques) dont nous disposons (patch-clamp, enregistrement extra-cellulaire, électro-encéphalogramme, ...) : l'exemple le plus éloquent est celui de la détection de l'émission d'un potentiel d'action. Quel système de mesure doit-on choisir pour détecter ce potentiel d'action et en mesurer l'amplitude, distinguer un *burst* (émission successive et rapprochée, en « bouffées » de potentiels d'action), d'un simple potentiel d'action ou tout simplement le distinguer du bruit dû aux conditions d'enregistrement ? Une fonction de coût telle que l'équation 1.22 où l'on remplace le courant ionique par la tension de membrane ne peut fournir des résultats intéressants que si l'optimisation est effectuée sur un intervalle de temps inférieur à la pseudo-période du signal à reproduire car ce type de fonction est très sensible aux déphasages temporels. En effet, réaliser l'estimation sur un intervalle plus long pourrait conduire l'algorithme à trouver un signal moyen, sans oscillation, considérant toute variation du potentiel de membrane comme du bruit, y compris les potentiels d'action. Nous reviendrons sur cette remarque au chapitre 4 où nous introduisons une méthode d'estimation des paramètres du modèle d'Hodgkin-Huxley à partir d'enregistrements de la tension de membrane sur un circuit électronique neuromimétique. Examinons donc les fonctions d'erreur employées, puis les méthodes d'optimisation associées, leurs avantages et leurs inconvénients.

### – La fonction de coût

Pour limiter la sensibilité au temps de la fonction de coût quadratique :

$$F_{cout} = \sum_t \sum_{stim} (V_{est}(stim, t) - V_{ref}(stim, t))^2, \quad (1.23)$$

la parade la plus couramment utilisée est la combinaison linéaire de plusieurs fonctions de coût, généralement des erreurs quadratiques, visant à satisfaire chacune un critère différent. Pour cela, Q. J. M. Huys et al. [HAP06] ont choisi de combiner les informations tirées des courants ioniques et des enregistrements de la tension de membrane en imposant non pas une stimulation constante, mais une stimulation pseudo-périodique pour synchroniser leur signal. Leur but est d'estimer les paramètres d'un modèle détaillé multi-compartimental. L'optimisation est scindée en sous-problèmes pour trouver séparément

les paramètres des synapses, des canaux de la membrane, ... D'autres auteurs, comme S. Druckmann ([DBG<sup>+</sup>07]) ou M. Pospischil et al. ([PTRM<sup>+</sup>08]) ont utilisé l'intervalle de temps entre le premier potentiel d'action et l'initiation de la dépolarisation, les intervalles entre le premier, le second et le dernier potentiel d'action. Des alternatives mono-critère insensibles à la temporalité des événements apparaissent également dans [RBD07], où la fonction de coût s'exprime directement en fonction de la fréquence, ou dans [LM01], où la fonction de coût s'exprime dans le plan de phase  $(V, dV/dt)$ . Elle est alors sensible aux variations dans le temps, mais pas à la ponctualité des événements. Ces dernières méthodes nécessitent d'avoir pré-réglé le modèle pour avoir des oscillations en réponse à une stimulation.

Quelques articles se sont attachés à la comparaison des performances de différentes fonctions de coût [KPK05], [VB99]). Keren et al. ont comparé les performances d'un même algorithme d'optimisation avec 5 fonctions différentes : la première est celle de l'équation 1.23, la deuxième est l'erreur quadratique entre les intervalles entre potentiels d'action (ISI : Inter Spike Interval), la troisième est l'erreur quadratique entre les plans de phases  $(V, dV/dt)$ , et les deux dernières sont des combinaisons des précédentes (ISI + eq. 1.23, puis somme des trois premières). Il ressort de cette étude que la fonction combinant les trois critères est celle fournissant les meilleurs résultats visuels entre les enregistrements et le modèle, avec un écart-type relativement faible par rapport aux solutions obtenues avec les autres fonctions. La fonction ISI est la moins adaptée à ce problème, la densité de trajectoire surestime la contribution des potentiels d'action et la combinaison ISI + eq. 1.23 reproduit mal les comportements passifs. La contrepartie à la qualité de la solution est évidemment le temps de convergence beaucoup plus lent dans le cas de la combinaison des trois.

Dans de nombreux cas on ne connaît pas le comportement analytique de la fonction de coût et seule l'expérimentation met en évidence l'existence de minima locaux ([AS06]). Il est donc fréquent que des métaheuristiques soient utilisées pour l'estimation des paramètres, à moins que la zone des paramètres recherchés ait été circonscrite et qu'une méthode d'optimisation locale suffise.

#### – Les méthodes de recherche associées

En plus de la comparaison des fonctions de coût, Vanier et al. ([VB99]) ont comparé différents algorithmes d'optimisation pour une même fonction sur 5 modèles de neurones (de 3 à 23 paramètres, mono- à multi-compartmentaux) : le gradient conjugué, les algorithmes génétiques, le recuit simulé et la recherche aléatoire. Sur une station de travail, l'estimation des 23 paramètres peut prendre jusqu'à un an<sup>7</sup>. Après lecture de cette études, nous pouvons retenir les remarques suivantes :

1. le gradient conjugué atteint toujours le minimum local le plus proche dans l'espace des paramètres et ne peut en sortir.
2. La recherche aléatoire peut conduire au minimum global, mais son temps de convergence est déraisonnable quel que soit le modèle.
3. Pour tous les modèles testés, les algorithmes génétiques et le recuit simulé fournissent les meilleurs résultats.
4. Pour les modèles comportant un faible nombre de paramètres (4 et 8) et où seules

<sup>7</sup>Il faut tout de même relativiser ce chiffre qui est lié aux performances des PC en 1999.

les conductances maximales sont estimées, le recuit simulé est meilleur en terme de valeurs obtenues et de temps de convergence.

5. Lorsque le nombre de paramètres est grand et que ceux-ci sont fortement indépendants les uns des autres (ce qui n’est pas le cas lorsqu’il s’agit d’estimer des conductances maximales et des constantes de temps associées par exemple), les algorithmes génétiques montrent leur supériorité en ce qui concerne à la fois la qualité de la solution et le nombre d’itérations nécessaires à la convergence.

Ces observations ont été par la suite corroborées par les publications d’auteurs utilisant tour à tour des méthodes de gradient ou simplexe ([TMM00], [RBD07], [HMLS07] et [TG08] pour des modèles de bursts), le recuit simulé ([PTRM<sup>+</sup>08]) et les algorithmes génétiques ([KPK05], [GPB05], [DBG<sup>+</sup>07]).

Finalement, l’engouement pour la modélisation des neurones a donné lieu à la création de bases de données de paramètres pour les modèles à conductances mono- ou multi-compartimentaux ([GGMA01], [PBM03], [THPM06]). Au lieu de proposer une méthode d’estimation des paramètres, les auteurs fournissent des bases de données répertoriant de manière la plus exhaustive possible les comportements de neurones en fonction de la valeur maximale des conductances, dans un espace de dimension 8 pour [THPM06] ou [PBM03] et dans un espace à 5 conductances pour [GGMA01]. Cependant, même avec ce type d’outil, se pose la question d’identifier les différentes caractéristiques du signal à reproduire afin d’extraire les paramètres adéquats.

## 1.3 Retour sur le formalisme d’Hodgkin–Huxley et la méthode du voltage-clamp

Comme annoncé dans la section précédente, le formalisme d’Hodgkin et Huxley reproduit, à l’échelle de la cellule, les comportements d’un neurone vivant. Nous détaillons ici ce formalisme, la technique de mesure du voltage-clamp, ainsi que la méthode d’estimation disjointe associée généralement utilisée pour déterminer les paramètres du modèle.

### 1.3.1 Le formalisme d’Hodgkin–Huxley : une analogie biologie/électricité

Après avoir mis en évidence le rôle des courants ioniques qui sous-tendent l’activité neuronale dans un axone géant de calmar ([HHK49]), Alan Lloyd Hodgkin et Andrew Fielding Huxley publièrent en 1952, dans un même volume du *Journal of Neurophysiology*, quatre articles décryptant les mécanismes du fonctionnement neuronal, [HHK52], [HH52b], [HH52a], [HH52c]. Une cinquième publication [HH52d] (citée plus haut), sortie la même année, vint compléter et faire la synthèse des quatre premières. Ces travaux, ainsi que ceux menés contemporanément par John Carew Eccles sur les mécanismes synaptiques, révolutionnèrent la compréhension du fonctionnement neuronal et de la communication nerveuse et leur valurent le prix Nobel de Médecine que tous trois se partagèrent en 1963.

#### Le formalisme original

Le formalisme qu’ils ont élaboré propose une analogie de comportement entre la membrane d’un neurone et le circuit électrique présenté à la figure 1.8. La membrane a été assimilée à un condensateur de capacité  $C_{MEM}$  et chaque canal ionique à une conductance variable  $g_{ion}$  de valeur

maximale  $g_{ion}^-$ , et un potentiel d'équilibre  $E_{ion}$ . Cette dernière analogie traduit la perméabilité variable aux différents ions relativement à la valeur du potentiel membranaire et les potentiels d'équilibre correspondent aux potentiels d'équilibre électrochimique propres à chaque espèce ionique.

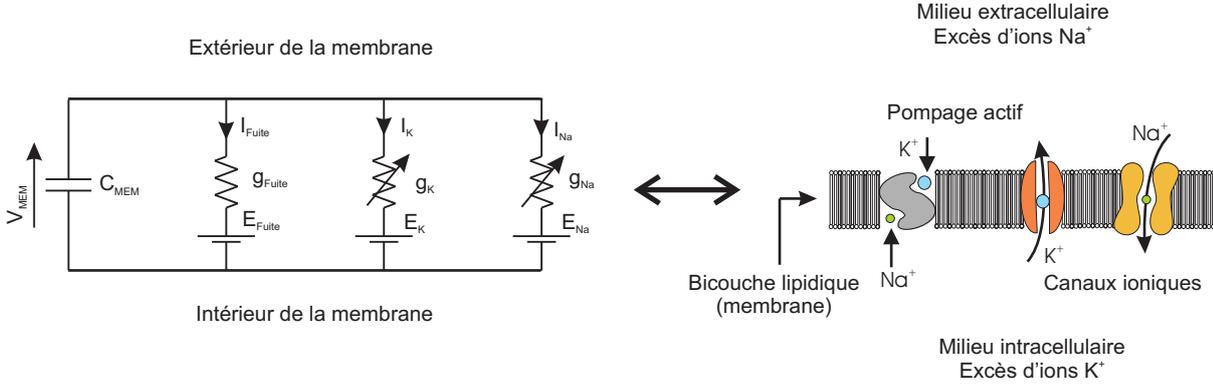


FIG. 1.8 – Modèle d'Hodgkin–Huxley : analogie biologie/électronique.

Mathématiquement, le fonctionnement du schéma électrique se traduit de la façon suivante :

$$-C_{MEM} \frac{dV_{MEM}(t)}{dt} = I_K(t) + I_{Na}(t) + I_{fuite}(t) \quad (1.24)$$

où  $V_{MEM}$  désigne le potentiel de membrane et  $K$ ,  $Na$ ,  $fuite$ , le potassium, le sodium et la fuite respectivement pour chaque grandeur considérée.

$I_{ion}$ ,  $i = K, Na, fuite$ , donné par l'équation 1.25, est le courant ionique.

$$I_{ion}(t) = g_{ion}(V_{MEM}(t)) \cdot (V_{MEM}(t) - E_{ion}) \quad (1.25)$$

où  $g_{fuite}$  est constant, mais  $g_K$  et  $g_{Na}$  sont variables.

La dépendance des conductances ioniques au potentiel de membrane a également été décrite dans les travaux d'Hodgkin et Huxley qui se fixèrent pour objectif d'exprimer mathématiquement les changements de perméabilité au  $K^+$  et au  $Na^+$ . La conductance membranaire est en effet définie comme l'inverse de la résistance membranaire ; elle est donc étroitement liée à la perméabilité. Leurs expériences les ont menés aux constatations suivantes, que nous avons déjà évoquées à propos de la physiologie du neurone. Premièrement, après une stimulation, il faut un certain délai avant que les différents canaux ioniques ne soient opérationnels ; on parle alors d'*activation*. L'activation de la conductance potassique met ainsi plusieurs millisecondes avant d'atteindre son maximum alors que celle de la conductance sodique ne prend au plus que quelques dizaines de microsecondes. De plus, contrairement à la conductance potassique commune, la conductance sodique possède la propriété de *s'inactiver* rapidement après son activation. Deuxièmement, le décours temporel de ces conductances est également sensible à la valeur de la tension : les conductances augmentent d'autant plus que le neurone est dépolarisé.

On sait maintenant que la perméabilité au sodium et au potassium dépend de l'état des sites moléculaires des protéines membranaires constituant les « canaux » qui peuvent être soit

ouverts, soit fermés. La probabilité de transition entre les deux états (ouvert/fermé) est définie par la fonction  $\beta$  (dépendante de la tension) pour le passage de l'état ouvert à l'état fermé, et par  $\alpha$ , pour le changement inverse. Chaque couple  $(\alpha, \beta)$  est spécifique de l'activation ou de l'inactivation pour un type d'ion donné. On associe généralement aux équations le diagramme suivant :



Ensuite, en posant  $x$  la fraction de canaux dans l'état ouvert et  $(1 - x)$  la fraction dans l'état fermé, nous pouvons définir la variation de la fraction de canaux ouverts comme la différence du produit de la fraction de canaux fermés par la probabilité d'ouverture et du produit de la fraction de canaux ouverts par la probabilité de fermeture. Autrement dit, nous pouvons écrire la formule suivante :

$$\frac{dx}{dt} = (1 - x)\alpha(V_{\text{MEM}}) - x\beta(V_{\text{MEM}}) \quad (1.27)$$

L'activation et l'inactivation peuvent ainsi être identifiées à la variable  $x$  de l'équation 1.27. Hodgkin et Huxley ont de plus montré expérimentalement que les termes d'activation/inactivation pouvaient être modulés par un exposant qui varie selon l'espèce ionique. Par convention, la variable d'activation du  $K^+$  est généralement appelée  $n$ , celle du  $Na^+$ ,  $m$ , et l'on désigne par  $h$  son terme d'inactivation. Les conductances ioniques correspondantes s'écrivent donc ainsi :

$$g_K(V_{\text{MEM}}(t)) = \bar{g}_K \cdot n^4(V_{\text{MEM}}(t)) \quad (1.28)$$

$$g_{Na}(V_{\text{MEM}}(t)) = \bar{g}_{Na} \cdot m^3(V_{\text{MEM}}(t)) \cdot h(V_{\text{MEM}}(t)) \quad (1.29)$$

d'où l'expression des courants obtenue en réinjectant dans l'équation 1.25 :

$$I_K(V_{\text{MEM}}(t)) = \bar{g}_K \cdot n^4(V_{\text{MEM}}(t)) \cdot (V_{\text{MEM}}(t) - E_K) \quad (1.30)$$

$$I_{Na}(V_{\text{MEM}}(t)) = \bar{g}_{Na} \cdot m^3(V_{\text{MEM}}(t)) \cdot h(V_{\text{MEM}}(t)) \cdot (V_{\text{MEM}}(t) - E_{Na}) \quad (1.31)$$

Notons finalement que pour plus de réalisme dans les modèles, il est possible, selon le comportement visé ou le type de neurone à modéliser, d'ajouter d'autres courants ioniques (calcique, potassium dépendant du calcium, ...) sur la base de l'analogie proposée par A. L. Hodgkin et A. F. Huxley. Le socle d'équations reste alors le même, mais des variantes peuvent apparaître dans les exposants des termes d'activation et d'inactivation selon le type d'ion considéré. La reproductibilité du comportement résidera bien entendu dans l'ajustement des paramètres du modèle : conductances maximales, constantes dans les expressions de  $\alpha$  et  $\beta$  (ci-dessous). A titre d'illustration, on pourra consulter l'article [PTRM<sup>+</sup>08] qui présente plusieurs variantes du modèle pour reproduire l'activité électrique de neurones thalamiques et corticaux.

### Modèle et variantes pour la cinétique de l'état d'équilibre et les termes d'activation et inactivation

A. L. Hodgkin et A. F. Huxley ont pu établir expérimentalement les équations des termes  $\alpha$  et  $\beta$  de chaque variable d'activation ou d'inactivation pour chaque canal ionique<sup>8</sup>. Ces modèles

<sup>8</sup>Une autre approche peut être faite en utilisant des modèles, aujourd'hui très répandus, issus de la thermodynamique [DH01].

fournissent les équations suivantes :

$$\alpha(V_{\text{MEM}}) = A.\exp\left(-\frac{\gamma.F.(V_{\text{MEM}} - V_{\text{offset}})}{R.T}\right) \quad (1.32)$$

$$\beta(V_{\text{MEM}}) = B.\exp\left(-\frac{(1-\gamma).F.(V_{\text{MEM}} - V_{\text{offset}})}{R.T}\right) \quad (1.33)$$

où :  $\gamma$  est une contante de  $[0; 1]$ ,

$F$  est la constante de Faraday ( $F = 9,64853.10^4 C.mol^{-1}$ ),

$R$  est la constante des gaz parfaits ( $R = 8,31447 J.mol^{-1}.K^{-1}$ ),

$T$  est la température en Kelvin

et  $V_{\text{offset}}$  est appelée tension de demi-activation.

En vue d'introduire une simplification du modèle proposée dans les autres thèses de l'équipe ([Dou], [Sai04]), définissons la constante de temps<sup>9</sup> (ou cinétique)  $\tau(V_{\text{MEM}})$  ainsi que le second membre  $x_{\infty}$  (appelé aussi état d'équilibre<sup>10</sup> ou régime permanent<sup>11</sup>) de l'équation différentielle 1.27. La première variable est homogène à un temps et caractérise la vitesse d'ouverture/fermeture des canaux, la seconde, sans dimension, représente la proportion de canaux ouverts en fonction de  $V_{\text{MEM}}$ , à  $V_{\text{MEM}}$  constant (comme dans le cadre des expériences de voltage-clamp qui seront décrites plus loin) :

$$x_{\infty}(V_{\text{MEM}}) = \frac{\alpha(V_{\text{MEM}})}{\alpha(V_{\text{MEM}}) + \beta(V_{\text{MEM}})} \quad (1.34)$$

$$\tau(V_{\text{MEM}}) = \frac{1}{\alpha(V_{\text{MEM}}) + \beta(V_{\text{MEM}})} \quad (1.35)$$

$$= \frac{x_{\infty}(V_{\text{MEM}})}{\alpha(V_{\text{MEM}})} \quad (1.36)$$

En remplaçant, dans l'équation 1.27,  $\alpha$  et  $\beta$  par leurs expressions en fonction de  $\tau(V_{\text{MEM}})$  et  $x_{\infty}(V_{\text{MEM}})$ , on obtient la relation :

$$x_{\infty}(V_{\text{MEM}}(t)) = \frac{1}{1 + \exp\left(\pm \frac{(V_{\text{MEM}}(t) - V_{\text{offset}})}{V_{\text{pente}}}\right)} \quad (1.37)$$

où le signe du terme entre parenthèses est négatif lorsque  $x$  désigne une activation et positif lorsqu'il caractérise une inactivation.  $V_{\text{offset}}$ , constante, est le terme de demi-activation – ou inactivation –, autrement dit  $x_{\infty}(V_{\text{MEM}}) = 0,5$  quand  $V_{\text{MEM}} = V_{\text{offset}}$ . Remarquons enfin que la constante  $V_{\text{pente}}$ , contrairement à son emploi mathématique habituel, ne désigne pas la tangente à la courbe au point d'inflexion  $V_{\text{offset}}$  ; il est seulement assimilé à une tension pour l'homogénéité de l'équation.

Afin de faciliter l'implantation des équations sur silicium, l'équipe a proposé de simplifier les équations précédentes en considérant que la fonction  $\tau$  était constante ([Dou], [Sai04]). Les

<sup>9</sup>Cette constante, qui n'est pas constante ici car dépend de  $V_{\text{MEM}}$ , est dénommée ainsi par les physiciens.

<sup>10</sup>Vocabulaire pour neuroscientifiques.

<sup>11</sup>Dénomination à l'usage des physiciens.

équations du second membre et de la cinétique ainsi retenues pour les premières implantations de neurones sur silicium dans l'équipe ont donc été l'équation 1.37 avec :

$$\tau_x \cdot \frac{dx(t)}{dt} = x_\infty(V_{\text{MEM}}(t)) - x(t) \quad (1.38)$$

où  $\tau_x$  est constant et spécifique à la fois de l'ion et de l'état (activation/inactivation). C'est cette formulation modifiée du formalisme d'Hodgkin–Huxley que nous utiliserons le plus souvent car les méthodes d'optimisation présentées dans cette thèse ont avant tout été testées et validées sur nos circuits électroniques. Nous rappellerons par la suite, chaque fois qu'il sera question d'ajuster un modèle, à quelle variante nous nous rapportons.

### 1.3.2 Le voltage-clamp et la méthode disjointe : limitations

#### Description de la technique du voltage-clamp

Le célèbre protocole du voltage-clamp ou voltage imposé toujours utilisé aujourd'hui par de nombreux neurophysiologistes a été proposé pour la première fois par Kenneth Cole et Georges Marmont dans les années quarante ([Col49], [Col55], [CM60], [Mar49]). Ce dispositif est appelé ainsi car il stabilise le potentiel de membrane à un niveau imposé par l'expérimentateur. Il permet de caractériser les propriétés d'activation et/ou d'inactivation d'un canal ionique. Dans la suite, nous appellerons "stimulation" le fait d'imposer une valeur de potentiel à la membrane.

Les enregistrements de voltage imposé nécessitent de traiter péalablement la membrane du neurone par l'injection de toxines permettant d'inhiber l'activité de certains canaux ioniques pour ne maintenir actif que le type de canal ionique dont on souhaite enregistrer l'activité. Par exemple, l'injection de tétrodoxine, un alcaloïde neurotoxique, bloque le courant sodique sans affecter le courant potassique. Inversement, le tétraéthylammonium bloque le potassium sans inhiber les canaux sodiques.

Le protocole de stimulation est différent selon qu'il s'agit de caractériser le processus d'activation ou le processus d'inactivation. Afin de caractériser l'activation d'un canal, un potentiel hyperpolarisé est d'abord appliqué à la membrane, ensuite cette hyperpolarisation est suivie par une série de dépolarisations comme à la figure 1.9 illustrant la méthode sur le courant potassique où sur la figure 1.10 relative au sodium.

Quant à caractériser l'inactivation, on impose en premier une série d'hyperpolarisations toutes suivies par un même potentiel dépolarisé ; la figure 1.11 illustre cette deuxième série de stimulations sur le courant sodique. Les stimulations sont imposées suffisamment longtemps ( $\gtrsim 30ms$ ) pour que le système soit considéré comme étant en régime permanent. Comme nous ne disposons pas d'enregistrements sur des neurones biologiques, les figures présentées dans cette section proviennent d'un circuit neuromimétique avec lequel il est possible d'effectuer des mesures en potentiel imposé. Pour retrouver des valeurs biologiques, il faut effectuer les conversions suivantes :  $V_{elec} = 5 \times V_{bio}$  et  $I_{elec} = 113,6 \times I_{bio}$ . En effet, pour limiter la valeur du rapport signal/bruit, les tensions du circuit ont été multipliées par 5 par rapport aux valeurs biologiques ; quant à la valeur du courant, elle est déduite de la relation  $g_{elec} = C_{elec}/C_{bio} \times g_{bio}$ .

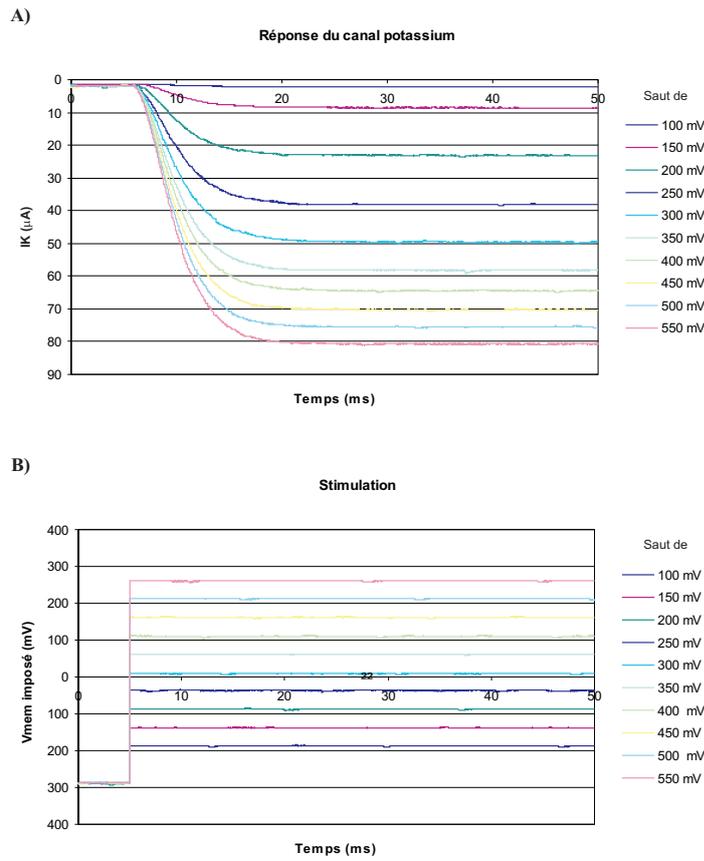


FIG. 1.9 – Illustration de la méthode du voltage-clamp sur le canal potassique. A) Réponse du courant potassique aux sauts de tensions de  $V_{MEM}$  du B). La valeur absolue du courant augmente avec  $V_{MEM}$ . B) Sauts imposés à  $V_{MEM}$ . (D'après [Sai04]), courbes obtenues à partir de circuits analogiques obéissant à une version approchée des équations de HH.

### Description de la méthode d'estimation disjointe associées au voltage-clamp

Après avoir enregistré les réponses des canaux ioniques aux stimulations, la méthode disjointe permet d'identifier un à un tous les paramètres du modèle d'Hodgkin–Huxley pour chaque canal ionique. Nous détaillons ici la méthode d'estimation sur un modèle à trois conductances (canaux de fuite, potassique et sodique).

Le canal potassique est facile à caractériser car il ne possède qu'un terme d'activation,  $n$  (équation 1.28). En se référant à la figure 1.9, quand  $t > 30ms$ , le courant atteint son régime permanent, soit  $n = n_\infty$ . Pour  $V_{MEM} > 300mV$  (valeur électrique), on observe une évolution linéaire (fig. 1.9B) du courant par rapport à l'évolution de la stimulation<sup>12</sup> : en régime permanent, les écarts entre les courbes de réponses sont identiques pour les stimulations les plus élevées (sauts de potentiels de 400, 450, 500 et 550 mV). En effet, pour les valeurs de potentiel imposé fortement positives, en observant l'équation 1.28, on peut déduire que  $n_\infty \simeq 1$  d'où, lorsque le

<sup>12</sup>Autrement dit, la réponse indicelle en régime permanent est la même que celle d'un premier ordre de gain statique constant, tandis que pour des échelons de plus faible amplitude, le gain statique varie avec l'amplitude de l'échelon de tension.

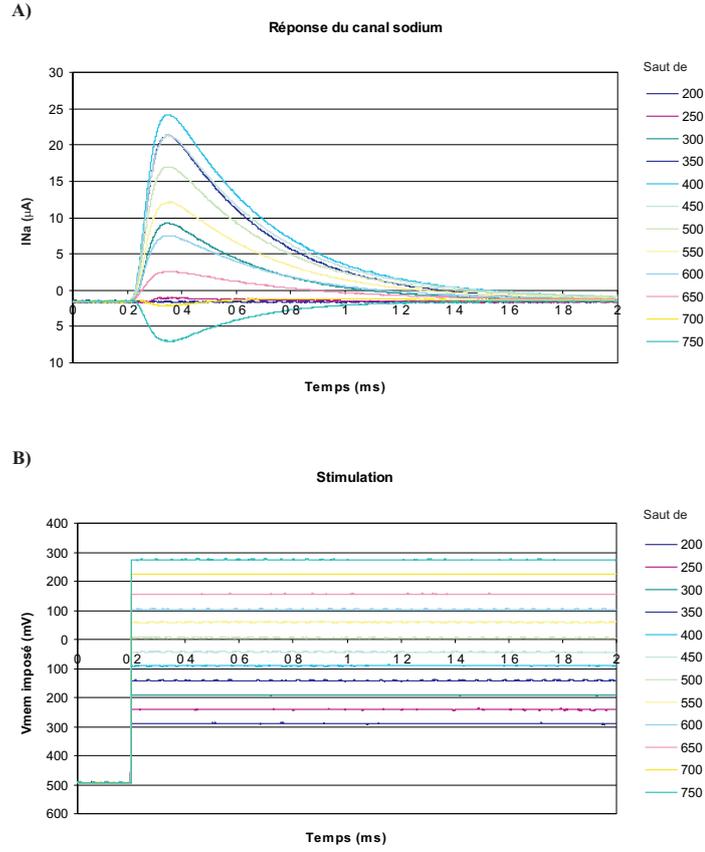


FIG. 1.10 – Illustration de la méthode du voltage-clamp sur le canal sodique – Caractérisation du terme d’activation. A) Réponse du courant sodique aux sauts de tensions de  $V_{MEM}$  du B). B) Sauts imposés à  $V_{MEM}$ . (D’après [Sai04]), courbes obtenues à partir de circuits analogiques obéissant à une version approchée des équations de HH

régime permanent est établi,  $n = 1$  et :

$$I_K = g_K \cdot (V_{MEM} - E_K) \quad (1.39)$$

En appliquant une régression linéaire aux courbes de la figure 1.9B pour  $t > 30ms$  et  $V_{MEM} > 300mV$ , on obtient  $E_K$  et  $g_k$ .

Puis, pour  $t > 30ms$ , on trace  $n_\infty$  en fonction de  $V_{MEM}$  à partir de l’équation 1.28. On obtient alors :

$$n_\infty = \left( \frac{I_K}{g_K(V_{MEM} - E_K)} \right)^{1/4} \quad (1.40)$$

La courbe qui en résulte est approchée par une sigmoïde avec  $V_{offset\_n}$  et  $V_{pente\_n}$  que l’on détermine souvent par une méthode de type descente de gradient.

Pour la constante de temps de l’activation  $\tau_n$ , on utilise une méthode d’approximation fréquente en physique :  $t = 3\tau_n$  à 81,5% de la valeur maximale. On utilise en effet l’expression temporelle du terme d’activation (en approchant l’équation de la décroissance temporelle

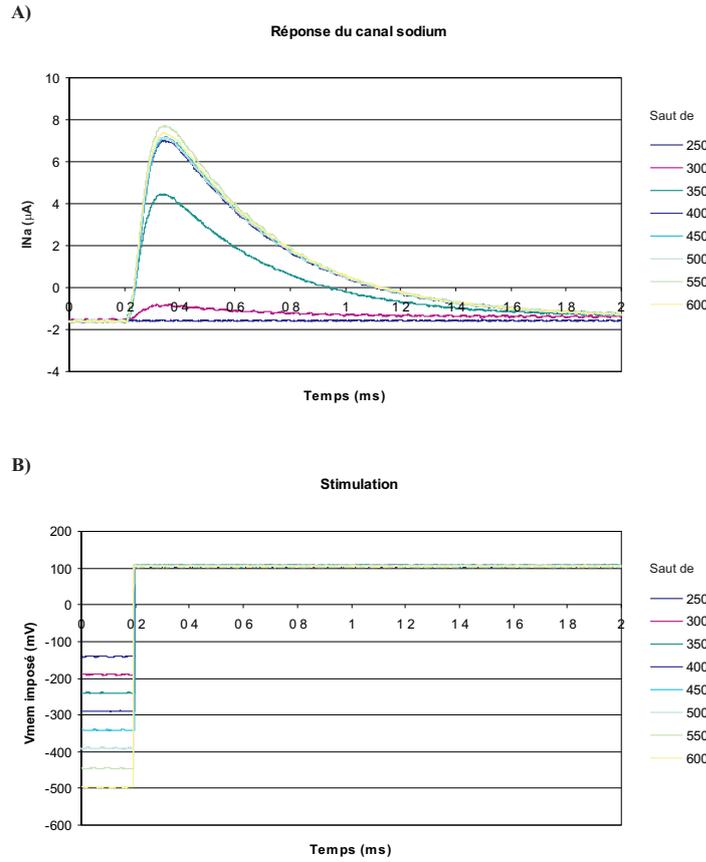


FIG. 1.11 – Illustration de la méthode du voltage-clamp sur le canal sodique – Caractérisation du terme d’inactivation. A) Réponse du courant sodique aux sauts de tensions de  $V_{MEM}$  du B). B) Sautes imposés à  $V_{MEM}$ . (D’après [Sai04]), courbes obtenues à partir de circuits analogiques obéissant à une version approchée des équations de HH

par une exponentielle) [HH52c] :

$$n = \left(1^{1/4} - 1^{1/4} \cdot \exp\left(\frac{-t}{\tau_n}\right)\right)^4 = \left(1 - \exp\left(\frac{-t}{\tau_n}\right)\right)^4 \quad (1.41)$$

Après injection de cette valeur particulière de  $n$  dans l’équation 1.28, on peut identifier  $\tau_n$  sur la courbe de la figure 1.9A. Cependant, cette technique d’évaluation de  $\tau_n$  reste très imprécise. La cinétique est parfois évaluée manuellement en faisant varier sa valeur dans un intervalle “réaliste” et en comparant visuellement le courant simulé avec le courant enregistré.

A la différence du canal potassique, le canal sodique ( $Na$ ) possède un terme d’activation,  $m$ , et un terme d’inactivation,  $h$  (équation 1.29). D’après des observations biologiques,  $\tau_m \ll \tau_h$ . Nous pouvons alors identifier séparément  $\tau_m$  et  $\tau_h$ .

La figure 1.10A montre la réponse en courant du canal sodique aux mêmes stimulations que celles appliquées figure 1.9A. L’estimation des paramètres  $E_{Na}$ ,  $g_{Na}$ ,  $V_{offset\_m}$ ,  $V_{pente\_m}$  et  $\tau_m$  ressemble à celle du canal potassique.

Nous supposons que la phase de croissance de la courbe-réponse du canal sodique correspond

à l'étape d'activation du canal et que l'inactivation n'entre pas encore en jeu dans cet intervalle de temps – puisque sa cinétique est beaucoup plus lente que celle de l'activation. Pour les stimulations correspondant aux plus grands sauts de potentiel imposé ( $-500mV$  et  $550$  à  $700mV$ ), figure 1.10B, nous émettons l'hypothèse que l'activation du canal sodique est complète. Nous considérons que lorsque le réseau de courbes atteint son maximum, le terme d'activation a atteint le régime permanent, ie  $m = m_\infty$ . Il s'en suit que la valeur maximale du courant ne dépend ni du terme d'activation, ni du terme d'inactivation. L'équation 1.29 fournit alors :

$$\max_t (I_{Na}(V_{MEM})) = g_{Na} \bar{n}_a (V_{MEM} - E_{Na}) \quad (1.42)$$

En traçant  $\max_t (I_{Na})$  par rapport à  $V_{MEM}$ , on peut ensuite trouver, comme pour le potassium,  $g_{Na}$  et  $E_{Na}$  par une régression linéaire.

Pour estimer les paramètres de l'activation  $V_{offset\_m}$  et  $V_{pente\_m}$ , on procède exactement comme pour le potassium : on trace la courbe  $m_\infty$  en fonction de  $V_{MEM}$  à partir de l'équation 1.45 que l'on approche par la sigmoïde de l'équation 1.37.

$$m_\infty = \left( \frac{\max_t (I_{Na}(V_{MEM}))}{g_{Na}(V_{MEM} - E_K)} \right)^{1/3} \quad (1.43)$$

Pour trouver la cinétique  $\tau_m$  on utilise l'expression temporelle du terme d'activation :

$$m = \left( 1^{1/3} - 1^{1/3} \cdot \exp\left(\frac{-t}{\tau_m}\right) \right)^3 = \left( 1 - \exp\left(\frac{-t}{\tau_m}\right) \right)^3 \quad (1.44)$$

Pour estimer les paramètres se rapportant au terme d'inactivation, on utilise la deuxième série de courbes (fig 1.11). La réponse en courant correspond cette fois-ci à l'activité de la proportion de canaux sodiques qui n'étaient pas initialement inactivés par le potentiel imposé. Comme le saut de potentiel est suffisamment élevé, l'activation est considérée complète dans la seconde partie de la réponse. Le courant maximal correspond donc à l'activation de la partie inactivée du sodium avant le saut de potentiel et on a :

$$h_\infty = \frac{\max_t (I_{Na}(V_{MEM}))}{g_{Na}(V_{MEM} - E_{Na})} \quad (1.45)$$

Le raisonnement employé pour estimer  $V_{offset\_m}$  et  $V_{pente\_m}$  et  $\tau_m$  est réutilisé ici pour déterminer  $V_{offset\_h}$  et  $V_{pente\_h}$  et  $\tau_h$ .

Il reste à estimer la valeur des paramètres correspondant au courant de fuite d'équation :

$$I_{fuite} = g_{fuite}(V_{MEM} - E_{fuite}) \quad (1.46)$$

$I_{pente\_m}$  dépendant linéairement de  $V_{MEM}$ , on obtient  $g_{pente\_m}$  et  $E_{pente\_m}$  par une simple régression linéaire d'après une série de mesures en faisant varier  $V_{MEM}$ .

### Limitations : quelques résultats d'estimation

Bien qu'encore très utilisée, la méthode d'estimation disjointe associée au voltage imposé fournit souvent des résultats erronés pour les constantes de temps, en particulier pour le canal sodique pour lequel elle peine aussi à évaluer la conductance maximale. Ce défaut est en grande partie dû aux assertions faites quant à l'indépendance de l'activation et de l'inactivation et

aux approximations sur les constantes de temps. Allan R. Willms fournit d'ailleurs une explication détaillée des faiblesses de cette technique dans [WBHWG99] et en propose une première alternative utilisant une méthode des moindres carrés non linéaire.

A titre d'illustration, nous présentons au tableau 1.2 les paramètres obtenus sur le canal sodique par la méthode d'estimation disjointe et par ajustement manuel des constantes de temps et conductance maximale : les figures 1.12 et 1.13 représentent respectivement les courbes obtenues pour la caractérisation de l'activation et de l'inactivation. Ici encore, des données provenant d'un circuit neuromimétique remplacent les données biologiques de référence. Pour comparaison, les simulations logicielles (courants « estimés » des figures 1.12 et 1.13) ont été effectuées avec le logiciel *Matlab*.

Paramètres	Méthode disjointe (a)	Ajustement manuel (b)	Variation en %  (a)-(b) /(b)
$g_{Na} (S)$	$16,0e^{-5}$	$9,46e^{-5}$	40,8%
$\tau_{m} (s)$	$4,80e^{-5}$	$2,95e^{-5}$	38,5%
$\tau_{h} (s)$	$4,00e^{-4}$	$3,60e^{-4}$	10,0%
$E_{Na} (V)$	$2,51e^{-1}$	$2,51e^{-1}$	0%
$V_{\text{offset}_m} (V)$	$-1,80e^{-1}$	$-1,80e^{-1}$	0%
$V_{\text{offset}_h} (V)$	$-2,16e^{-1}$	$-2,16e^{-1}$	0%
$V_{\text{pente}_m} (V)$	$3,46e^{-2}$	$3,46e^{-2}$	0%
$V_{\text{pente}_h} (V)$	$2,51e^{-2}$	$2,51e^{-2}$	0%

TAB. 1.2 – Paramètres estimés du modèle d'Hodgkin–Huxley pour le canal sodique. Valeurs estimées par la méthode disjointe et ajustement manuel ; variation des valeurs estimées.

On peut donc constater que l'erreur d'estimation fournie par cet exemple est de la moitié de la valeur de la conductance maximale. L'erreur sur la constante de temps est de 40% pour l'activation et de 10% environ pour l'inactivation. Cette constatation remet en question la fiabilité de la méthode d'estimation disjointe associée au voltage-clamp. La mauvaise qualité de l'estimation peut en fait s'expliquer par les approximations faites sur les termes d'activation et d'inactivation du sodium : on considère en effet ces deux termes indépendants l'un de l'autre alors qu'en réalité, ils interagissent fortement. De plus, l'hypothèse concernant l'activation totale des canaux ( $m = 1$ ) en présence d'un saut de tension élevé est trop forte, or cette supposition permet d'estimer la conductance maximale et le potentiel d'équilibre du canal ionique. Ceci affecte donc de manière importante la qualité de l'estimation de la conductance maximale.

Le réglage manuel du modèle d'Hodgkin–Huxley est très complexe et très coûteux en temps du fait de la non-linéarité des équations, de la quantité de paramètres et de leur sensibilité. Il est donc nécessaire, pour ajuster le modèle, de recourir à des méthodes automatiques d'estimation. Les alternatives à la méthode disjointe préalablement proposées (voir section 1.2.2) sont gourmandes en ressources calculatoires et tombent parfois dans le piège des minima locaux. Nous avons donc souhaité développer une nouvelle méthode d'estimation capable de passer outre ces problèmes, mais permettant de trouver les paramètres en quelques minutes voire quelques heures. Cette technique permettrait par ailleurs de régler les circuits neuromimétiques conçus par l'équipe. Les métaheuristiques semblent pouvoir répondre à nos attentes.

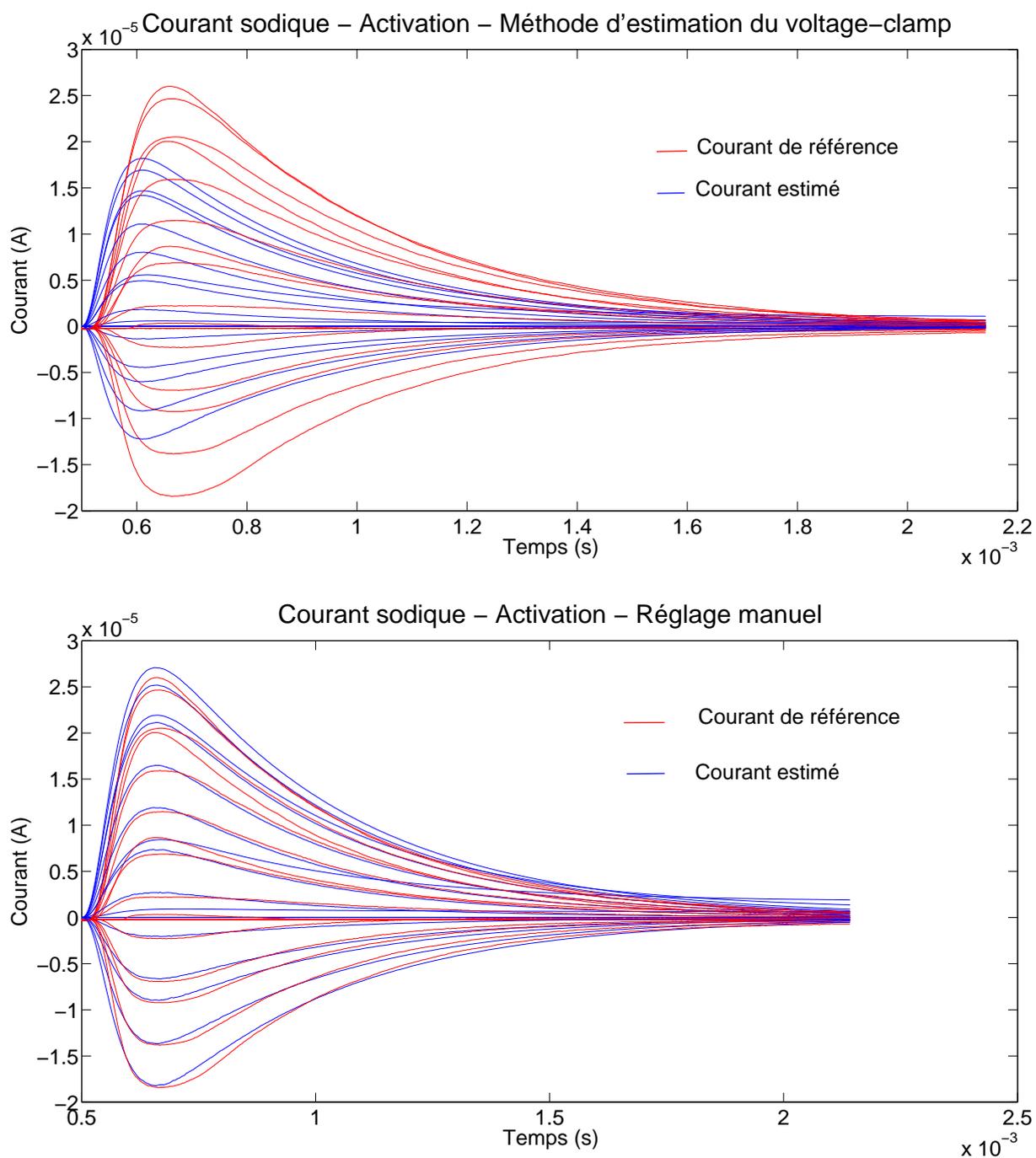


FIG. 1.12 – Courants sodiques estimés par la méthode d'estimation du voltage-clamp (haut) et manuellement (bas) – Réponses à la série de stimulations pour la caractérisation des variables d'activation.

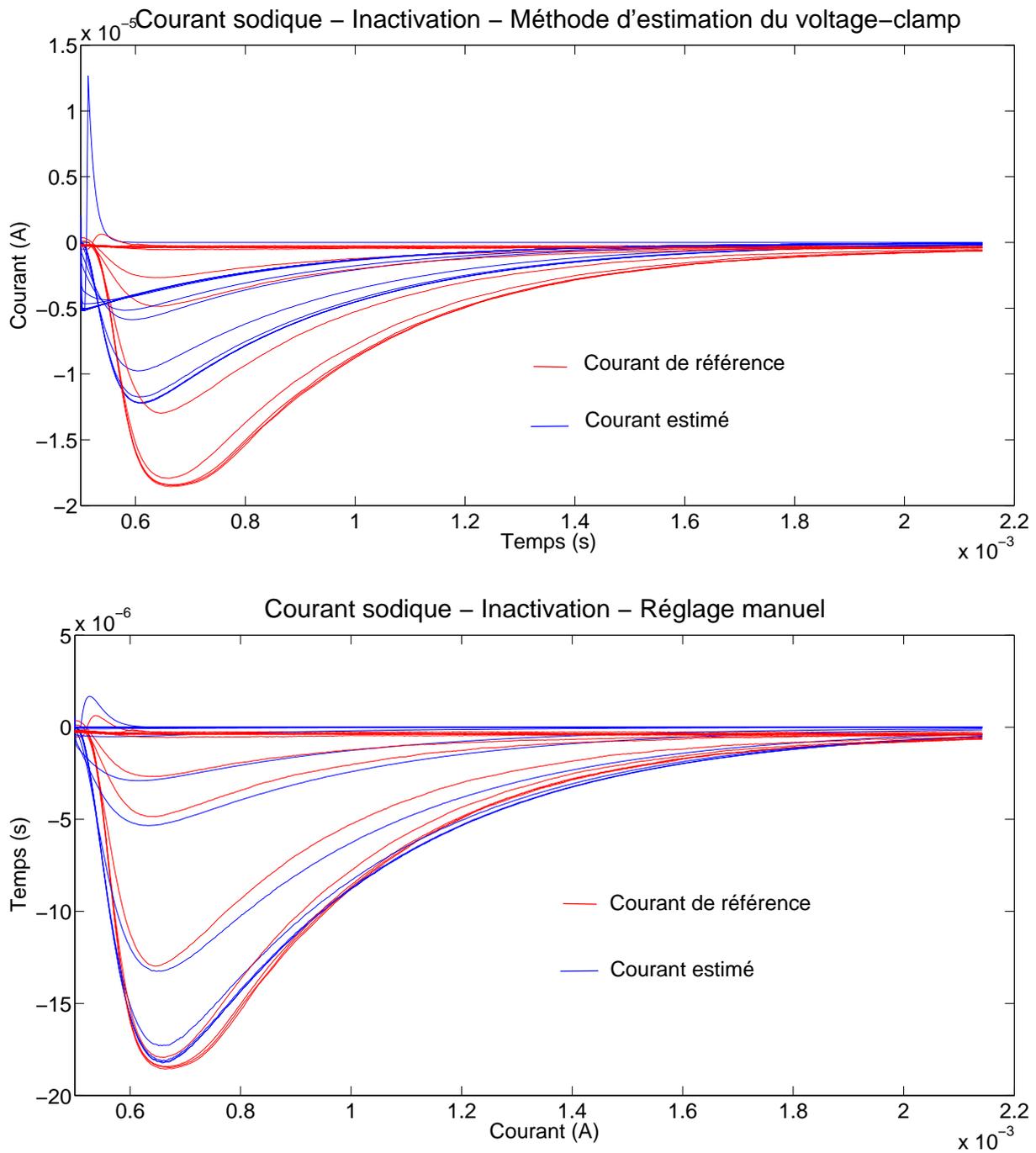


FIG. 1.13 – Courants sodiques estimés par la méthode d'estimation du voltage-clamp (haut) et manuellement (bas) – Réponses à la série de stimulations pour la caractérisation des variables d'inactivation.

## 1.4 Résumé

Ce chapitre présente les notions de base nécessaires à la compréhension de l'estimation de paramètres de modèles de neurones ainsi qu'un état de l'art des modèles et techniques d'optimisation. Après quelques préliminaires sur l'anatomie et la physiologie du neurone, nous abordons la problématique de l'estimation de paramètres de modèles de neurones en présentant les modèles les plus connus de la communauté, du neurone Integrate-and-Fire aux modèles à conductances multicompartimentaux. Nous évoquons ensuite le problème de l'automatisation de l'estimation de paramètres et décrivons les méthodes d'optimisation utilisées pour venir à bout de sa complexité. Pour finir, comme les circuits neuromimétiques conçus par l'équipe s'appuient sur le formalisme d'Hodgkin-Huxley, nous nous focalisons sur celui-ci, détaillant la méthode du voltage imposé et la technique d'estimation associée pour mettre en évidence ses limitations. Nous pouvons ainsi justifier de la nécessité de trouver une alternative aux méthodes d'estimation existantes.



## Chapitre 2

# Estimation des paramètres d'un modèle à conductances canal ionique par canal ionique

### Sommaire

---

<b>2.1</b>	<b>Le choix de la méthode d'estimation : comparaison de métaheuristiques . . . . .</b>	<b>38</b>
2.1.1	Choix d'une fonction d'erreur . . . . .	38
2.1.2	Protocole expérimental et implémentation des équations du modèle . . .	39
2.1.3	Comparaison de méthodes d'optimisation . . . . .	40
2.1.4	Validation numérique de la méthode du DE . . . . .	52
<b>2.2</b>	<b>Amélioration de l'algorithme d'Évolution Différentielle . . . . .</b>	<b>58</b>
2.2.1	Introduction de nouvelles variantes de l'évolution différentielle . . . . .	58
2.2.2	Étude comparative sur un benchmark de fonctions . . . . .	60
2.2.3	Application des variantes au canal potassique du modèle d'Hodgkin-Huxley . . . . .	63
<b>2.3</b>	<b>Résumé . . . . .</b>	<b>65</b>

---

« La science ne cherche pas à énoncer des vérités éternelles ou de dogmes immuables ; loin de prétendre que chaque étape est définitive et qu'elle a dit son dernier mot, elle cherche à cerner la vérité par approximations successives. »  
(Bertrand Russell dans *ABC de la relativité*)

L'équipe AS2N, anciennement ISN, a conçu des circuits analogiques neuromimétiques paramétrables s'appuyant jusqu'ici sur le formalisme d'Hodgkin et Huxley décrit dans le chapitre 1. L'un des enjeux de ce travail de thèse, outre proposer un outil de modélisation pour les neurophysiologistes était de concevoir pour l'équipe un système de réglage automatique des circuits électroniques neuromimétiques. Nous avons donc cherché, dans un premier temps, à estimer les paramètres de ce modèle pour reproduire une activité neuronale donnée. Comme nous l'avons vu au chapitre précédent, la méthode d'estimation du voltage-clamp nécessite des approximations qui peuvent entraîner d'importantes erreurs quant à la valeur des paramètres estimés. Pour illustrer cela, les figures 1.12 et 1.13 présentaient les courbes théoriques et les courbes obtenues avec les paramètres estimés par la technique d'estimation du voltage-clamp et par un réglage manuel (tableau 1.2).

La première étape a consisté à implémenter des équations du modèle. Pour trouver et tester notre future méthode d'estimation, nous avons tout d'abord choisi d'utiliser l'environnement *Matlab* qui possède des fonctions de haut niveau, permet de manipuler facilement les matrices sans avoir à gérer la mémoire de l'ordinateur, et donc de se concentrer sur la technique d'optimisation à mettre en oeuvre. L'implémentation des équations aux dérivées partielles du modèle nécessite une discrétisation : les dérivées  $\tau_x \frac{dx(t)}{dt}$  de l'équation 1.38 sont approchées par une méthode d'Euler (équation 2.1).

$$\tau_x \frac{x(t+1) - x(t)}{\Delta t} = x_\infty(t) - x(t) \quad \text{avec } x = n, m, h. \quad (2.1)$$

Comme décrit dans le premier chapitre, l'élaboration d'une méthode d'optimisation, après le choix du modèle et des données de référence, nécessite la sélection d'une fonction de coût et d'un algorithme d'optimisation. Ici, le choix du formalisme d'Hodgkin-Huxley était déterminé par les besoins de l'équipe. Le modèle retenu dans un premier temps est donc celui décrit par les équations 1.24, 1.30, 1.31, 1.37 et 1.38. La démarche conduisant à la sélection de la fonction de coût et de l'algorithme d'optimisation est décrite à la section 2.1. Nous avons fait deux sortes de validation de la méthode retenue : la première avec à des données fournies par un circuit analogique neuromimétique, la deuxième à partir de données obtenues par résolution numérique des équations du modèle. Nous avons ensuite introduit, dans la section 2.2, des variantes de l'algorithme d'évolution différentielle pour lequel nous obtenions les meilleurs résultats. Ces variantes visaient à améliorer les performances de l'algorithme.

## 2.1 Le choix de la méthode d'estimation : comparaison de métaheuristiques

Cette section présente l'élaboration du système d'estimation des paramètres du formalisme d'Hodgkin-Huxley, depuis le choix de la fonction d'erreur, jusqu'au choix de l'algorithme d'optimisation, en passant par la description du protocole expérimental. Afin de sélectionner l'algorithme le plus adéquat, nous avons comparé plusieurs métaheuristiques et proposé des variantes hybrides de celles-ci.

### 2.1.1 Choix d'une fonction d'erreur

La courbe des signaux à reproduire ne présente pas de périodicité. Nous avons donc choisi de minimiser la norme  $\mathcal{L}^2$  de la différence entre le signal d'origine et le signal estimé. Autrement

dit, pour chaque type d'ion, nous cherchons à minimiser<sup>13</sup> :

$$F_{\text{coût}}(x_{\text{ion}}) = \sum_t \sum_{\text{stim}} ((I_{\text{reference,ion}}(t, \text{stim}) - I_{\text{estime}}(x_{\text{ion}}, t, \text{stim}))^2) \quad (2.2)$$

où  $t$  désigne le temps,  $x_{\text{ion}}$  représente un vecteur de paramètres à l'instant  $t$ ,  $\text{ion}$  le type d'ion considéré ( $K$  ou  $Na$ ) et  $\text{stim}$  correspond aux différents sauts de tension imposés à la membrane du neurone. Dans la suite, lorsque l'on traitera d'un algorithme d'optimisation à population, on notera souvent  $F_{\text{fit}}(X_{\text{ion}})$ , où  $X_{\text{ion}}$  représente une matrice dont les lignes sont les vecteurs de paramètres  $x_{\text{ion}}$ .

Cependant, en raison des ordres de grandeur des courants ioniques manipulés (de l'ordre de  $10^{-6}A$ ), les valeurs de la fonction à minimiser étaient au plus de l'ordre de  $10^{-11}A^2$  et l'on atteignait les limites numériques de la précision du logiciel de calcul. Nous avons alors modifié la fonction ci-dessus en multipliant la différence des valeurs des courants ioniques par un facteur  $\lambda = 10^6$  :

$$F_{\text{coût}}(x_{\text{ion}}) = \sum_t \sum_{\text{stim}} (\lambda \times (I_{\text{reference,ion}}(t, \text{stim}) - I_{\text{estime}}(x_{\text{ion}}, t, \text{stim}))^2) \quad (2.3)$$

### 2.1.2 Protocole expérimental et implémentation des équations du modèle

Afin d'anticiper au mieux les performances des algorithmes d'optimisation sur des signaux vivants ou sur le réglage automatique de circuits neuromimétiques, nous souhaitons utiliser des signaux bruités. En effet, si nous avons débuté cette étude en prenant comme signal de référence des données logicielles (calculées à partir des équations du modèle), le fait qu'un algorithme d'optimisation fournisse un résultat d'estimation acceptable ne nous aurait pas nécessairement assuré d'obtenir des résultats comparables sur des données expérimentales, donc bruitées. Comme nous ne disposons pas d'enregistrements de voltage-clamp effectués dans le vivant, de même qu'au chapitre 1, nous avons utilisé comme signaux de référence des enregistrements de type voltage-clamp réalisés sur un circuit électronique neuromimétique développé dans l'équipe<sup>14</sup>. Il s'agissait ensuite de simuler logiciellement les différents courants ioniques et de les comparer à ceux obtenus matériellement (sur la puce). Notons cependant qu'il n'a pas été démontré que le bruit biologique était similaire au bruit des circuits intégrés. Pour chaque estimation, nous disposons, pour le potassium, d'enregistrements correspondant à 14 sauts de tensions imposées, contenant chacun entre 700 et 800 points; pour le sodium, nous avons utilisé deux types de séries de sauts (activation et inactivation), la première constituée d'un réseau de 16 sauts de tensions, la seconde d'un réseau de 8 contenant chacune 500 points environ.

Le principe d'estimation des paramètres du modèle est le suivant :

1. Pour commencer, l'initialisation de l'optimisation s'effectue par tirage aléatoire d'un jeu de vecteurs.
2. On "injecte" ensuite ces valeurs dans les équations des courants ioniques du modèle (équations 1.30 pour le potassium, équ. 1.31 pour le sodium, puis 1.37 et 1.38 communes aux deux

<sup>13</sup>On remarquera que dans cette notation, le courant de référence  $I_{\text{reference,ion}}$  n'est pas fonction de  $x_{\text{ion}}$  puisqu'il s'agit normalement d'enregistrements expérimentaux et non de courants ioniques calculés à partir d'un vecteur de paramètres.

<sup>14</sup>Notons que nous ne pouvions pas connaître les paramètres théoriques de référence car, en raison des disparités dues aux méthodes de fabrication des circuits (cf chapitre 3), tous les circuits ne répondent pas de la même façon à un jeu identique de paramètres et ont un comportement différent, voire très éloigné, de celui que fournirait la résolution numérique des équations du modèle avec les mêmes paramètres.

canaux ioniques) pour comparaison des courants ainsi calculés avec les enregistrements provenant de la puce (référence). Chaque canal ionique est traité séparément.

3. Après cette comparaison, on sélectionne des vecteurs de paramètres à conserver pour l'itération suivante (selon l'algorithme d'optimisation retenu).
4. On y applique des transformations relatives au choix de l'algorithme d'optimisation.
5. Puis l'on compare de nouveau avec les enregistrements de la puce (étape 2), et ainsi de suite jusqu'à atteindre le critère d'arrêt (nombre maximal d'itérations ou précision sur la valeur de la fonction de coût).

### 2.1.3 Comparaison de méthodes d'optimisation

Dans cette section, nous présentons les différents algorithmes d'optimisation que nous avons implémentés et comparés lors de l'étude préalable à l'application au réglage d'un système neuromimétique. Afin de se prémunir au mieux des problèmes de minima locaux, nous avons centré notre étude sur les métaheuristiques : recuit simulé, algorithmes génétiques, essais particuliers et évolution différentielle. Toutes, exceptées le recuit simulé, sont des métaheuristiques à population. Nous avons également, au cours de nos pérégrinations, tenté d'introduire des variantes hybrides de ces algorithmes en combinant par exemple l'évolution différentielle et le recuit simulé ou en adaptant une méthode utilisée en filtrage particulière. Comme nous ne maîtrisons pas, au démarrage des travaux, la précision des paramètres et n'avions donc pas pu définir de critère d'arrêt relatif à la précision exigée, la qualité des solutions fut évaluée, pour la comparaison, par des critères d'analyse visuelle des courbes. Notons au passage que selon le *No Free Lunch Theorem*, énoncé en 1997 par D. H. Wolpert et W. G. Macready dans [WM97], aucun n'algorithme d'optimisation n'est plus adapté que les autres pour résoudre tous les types de problèmes. Chaque problème nécessite une étude approfondie et un algorithme spécifique pour être résolu. Les algorithmes d'optimisation, même les plus utilisés (comme les algorithmes génétiques ou le recuit simulé), ne peuvent pas marcher de façon générale : ils ne seront efficaces que sur des problèmes avec des structures mathématiques bien spécifiques.

Les intervalles de recherche des paramètres étaient les mêmes pour chaque méthode testée et sont synthétisés dans le tableau 2.1. Notons respectivement  $D_K$  et  $D_{Na}$  les espaces de recherches associés au potassium et au sodium ;  $d_k$  et  $d_{Na}$  sont leurs dimensions respectives. Dans la suite, nous notons  $X_{min}$  (resp.  $X_{max}$ ) le vecteur (ou matrice) des bornes min (resp. max). Dans les algorithmes d'optimisation qui vont suivre, lorsqu'au cours de la recherche un point  $X_k$  "sort" de l'espace défini, il est automatiquement ramené à la borne la plus proche. Ainsi, pour redéfinir un nouveau vecteur  $x$  à partir d'un vecteur  $x_K$ , nous prenons toujours<sup>15</sup>  $x = \min(\max(x_k, X_{min}), X_{max})$ . Cette astuce permet en outre de détecter si l'espace de recherche imposé est trop restreint et d'agrandir les intervalles si nécessaire.

L'ensemble des simulations a été réalisé à l'aide de la version 7.7.0 du logiciel *Matlab* sur un PC *Dell* de processeur Intel Core duo à 2.66 GHz comportant 3 Go de RAM. Les algorithmes ont d'abord été testés uniquement sur le canal potassique. En effet, le comportement du canal sodique étant plus complexe, il aurait été inutile de tester sur le sodium une méthode ne fournissant pas de résultats satisfaisants sur le potassium. Les premiers résultats présentés concernent donc ce dernier.

<sup>15</sup>Le min de deux vecteurs  $a = (a_1, a_2, \dots, a_n)$  et  $b = (b_1, b_2, \dots, b_n)$  est ici défini par :  $\min(a, b) = (\min(a_1, b_1), \min(a_2, b_2), \dots, \min(a_n, b_n))$ , de même pour le max.

Paramètres	Potassium				Sodium			
	Borne min		Borne max		Borne min		Borne max	
	Élec	Bio	Élec	Bio	Élec	Bio	Élec	Bio
$g_{ion}$ ( $\mu S$ et $\mu S/cm^2$ )	$10^{-1}$	$9,68.10^{-7}$	100	$9,68.10^{-4}$	$10^{-1}$	$9,68.10^{-7}$	1000	$9,68.10^{-3}$
$\tau_{act}$ (ms)	0.1	0.1	10	10	0.02	0.02	10	10
$E_{equi}$ (mV)	-750	-130	0	0	20	4	600	120
$V_{offset\_act}$	-500	-100	0	0	-600	-120	0	0
$V_{pente\_act}$	-1	-0.2	200	40	1	0.2	500	100
$\tau_{inact}$					0.1	0.1	100	100
$V_{offset\_inact}$					-600	-120	0	0
$V_{pente\_inact}$					1	0.2	500	100

TAB. 2.1 – Intervalles de recherche des paramètres pour les canaux sodiques et potassique avec les valeurs biologiques et les valeurs du circuit électronique neuromimétique. Les valeurs des conductances maximales sont en  $\mu S$  pour les valeurs électriques et en  $\mu S/cm^2$  pour les valeurs biologiques.

### Le recuit simulé

Nous avons, pour commencer, orienté nos choix d'algorithmes suivant les méthodes présentées au chapitre précédent et déjà utilisées en modélisation neuronale. Nous avons retenu celles qui semblaient les plus performantes pour ce type de problèmes, ainsi le premier algorithme que nous avons testé fut le recuit simulé (voir Algorithme 2). L'algorithme en lui-même fut introduit pour la première fois par S. Kirkpatrick et al. dans [KGJV83].

La "bonne" convergence du recuit simulé – indépendamment de la fonction de coût – est conditionnée par l'ajustement de la température initiale,  $T_{init}$ , de la température finale,  $T_{fin}$ , du nombre d'itérations à chaque palier de température et de sa fonction d'évolution,  $h : T \rightarrow h(T)$ . Dans un premier temps,  $h$  a été définie comme fonction linéaire de la température :  $h(T) = K.T$  où  $K$  était égal à 0,95 afin de faire varier lentement la température et d'éviter de "geler" trop vite la solution au risque de rester dans un minimum local. Une fois  $h$  fixée, nous avons fait varier  $T_{init}$  dans l'intervalle [100; 10000] et  $T_{fin}$  dans [ $10^{-9}$ ;  $5.10^{-3}$ ]. Les meilleurs résultats ont été obtenus avec  $T_{init} = 1000$  et  $T_{fin} = 10^{-9}$ . Nous avons par ailleurs arbitrairement fixé égal à 100 le nombre d'itérations par palier de température. Le premier vecteur de paramètres a été initialisé aléatoirement à l'intérieur de l'espace de recherche.

Dans ces conditions, en moyennant les résultats obtenus avec 2 simulations<sup>16</sup>, l'estimation des paramètres prend environ 6 jours et 6 heures et l'erreur finale relative<sup>17</sup>,  $f_{opt}$ , est de l'ordre de  $6,7.10^{-4}$ . Les différents paramètres estimés sont regroupés, à titre de comparaison avec d'autres méthodes, dans le tableau 2.1. Les tracés des courbes de courant apparaissent quant à eux à la figure 2.1.

### Les algorithmes génétiques

Ces algorithmes, dont les bases théoriques furent introduites par J. Holland en 1975 dans [Hol75], ont été présentés de manière générale dans le premier chapitre. A l'instar du recuit simulé, ce type d'algorithme présentait de bons résultats quant à l'estimation de paramètres de modèles neuronaux, en particulier lorsque ceux-ci étaient fortement indépendants les uns des

<sup>16</sup>Compte tenu du temps d'exécution d'une estimation, nous n'avons pas pu en réaliser davantage.

<sup>17</sup>où  $f_{opt} = F_{coût}(X_{est}) / (\sum_t \sum_{stim} (10^6 \times I_{ion, ref}(t, stim)^2))$

autres. Bien que les constantes de temps du modèle ne soient pas indépendantes, nous avons testé cet algorithme sur notre problème. Nous détaillons notre implémentation dans Algorithme 4. Dans cette méthode, nous proposons de renouveler totalement par tirage aléatoire uniforme une proportion  $Q_{renouv}$  des moins bons individus de la population, c'est-à-dire que nous renouvelons les  $[N_{ind} \times Q_{renouv}]$  individus dont l'évaluation de la fonction de coût est la plus élevée. Nous conservons intacts – sans mutation, ni croisement – les 5 meilleurs individus de chaque itération.

---

**Algorithme 4** : Algorithmes génétiques – Implémentation testée.
 

---

**Entrées** :  $N_{iter}$  : nombre d'itérations,  $F_{coût}$  : fonction de coût,  $pc$  : probabilité de croisement,  $pm$  : probabilité de mutation,  $Nc$  : nombre de gènes croisés,  $Nm$  : nombre de gènes mutés,  $N_{ind}$  : nombre d'individus,  $Q_{renouv}$  : proportion d'individus renouvelés,  $mut_{ampl}$  : vecteur amplitude de la mutation autorisée de dimension  $d_k$ ).

**Sorties** :  $x_{opt}$ , minimisant  $F_{coût}$ .

**Initialisation** :  $X = X_0 = (x_1, x_2, \dots, x_{N_{ind}})$  matrice aléatoire dans  $D_K^{N_{ind}}$ ,  $x_{opt}$

Évaluer  $f(X_0)$

Classer par ordre croissant ses éléments

**pour**  $k$  de 0 à  $(N_{iter} - 1)$  **faire**

    À partir de la deuxième itération : renouveler les  $[N_{ind} \times Q_{renouv}]$  individus les moins bons dans la matrice  $X$

    On obtient une nouvelle matrice de paramètres  $X = \min(\max(X, X_{min}), X_{max})$

    Évaluer  $F_{coût}(X)$

    Classer par ordre croissant ses éléments

    Conserver les 5 meilleurs éléments et tirer de manière aléatoire uniforme  $[N_{ind}/2]$  individus à muter parmi les autres

    Pour chaque individu tiré, choisir de manière aléatoire uniforme  $Nm$  gènes à muter

    Effectuer la mutation dans  $mut_{ampl}$

    On obtient une nouvelle matrice de paramètres  $X = \min(\max(X_k, X_{min}), X_{max})$

    Évaluer  $F_{coût}(X)$

    Classer par ordre croissant ses éléments

    Conserver les 5 meilleurs éléments et tirer de manière aléatoire uniforme  $[N_{ind}/2]$  individus à croiser parmi les autres et associer des paires

    Pour chaque couple choisi, tirer de manière aléatoire uniforme  $Nc$  gènes à échanger

    Effectuer le croisement

    On obtient une nouvelle matrice de paramètres  $X = \min(\max(X_k, X_{min}), X_{max})$

    Évaluer  $F_{coût}(X)$

    Classer par ordre croissant ses éléments

$x_{opt} = x \in \{x_1, x_2, \dots, x_{N_{ind}}\}$  t.q.  $\forall x_i \in \{x_1, x_2, \dots, x_{N_{ind}}\}, F_{coût}(x) \leq F(x_i)$

retourner  $x_{opt}$

---

La qualité de la solution obtenue avec les algorithmes génétiques dépend de l'ajustement des paramètres de la méthode : le nombre d'itérations, le nombre d'individus de la population initiale, le nombre d'individus totalement renouvelés, la probabilité de croisement, la probabilité de mutation, le nombre de paramètres mutés et croisés, la taille de l'intervalle au sein duquel est effectuée la mutation. Dans notre algorithme, la proportion d'individus auxquels nous faisons

Paramètre	Valeur
$N_{iter}$	300
$N_{ind}$	300
$pc$	0,7
$pm$	0,1
$Nc$	1
$Nm$	1
$Q_{renouv}$	1/4

TAB. 2.2 – Hyperparamètres retenus pour l'algorithme génétique appliqué au canal potassique.

subir croisements et mutations est d'1/2 à chaque fois.

Afin de tester cette méthode, nous avons fait varier différents hyperparamètres. Bien que l'étude ne fût pas exhaustive, elle nous permet de retenir une configuration particulière. Les possibilités préalablement envisagées sont résumées ci-dessous :

- À  $N_{iter}$  et  $N_{ind}$  fixé, nous avons fait varier les probabilités de croisement,  $pc$ , et de mutation,  $pm$ , tout en autorisant la mutation et le croisement d'un seul gène à la fois.
- À  $N_{iter}$ ,  $N_{ind}$ ,  $pc$  et  $pm$  fixés, nous avons modifié le nombre de paramètres mutés ou croisés.
- Après avoir choisi les valeurs de  $pc$ ,  $pm$  et fixé à 1 le nombre de gènes mutés (resp. croisés) sur chaque individu désigné pour la mutation (resp. le croisement), nous avons fait varier le nombre d'itérations  $N_{iter}$  et le nombre d'individus  $N_{ind}$ .

Après divers essais, nous nous sommes arrêtés sur le choix d'hyperparamètres décrit au tableau 2.2 pour la caractérisation du canal potassique. Avec ce jeu d'hyperparamètres, l'estimation dure environ 1 h 55 min et l'erreur finale relative est de  $2,0 \cdot 10^{-3}$ .

## L'évolution différentielle

L'évolution différentielle (que nous noterons par la suite DE en référence à l'acronyme anglais) appartient, tout comme les algorithmes génétiques, à la famille des algorithmes évolutionnaires. Elle fut introduite par R. Storn et K. Price dans [SP97] en 1997. Elle est aujourd'hui utilisée dans différents domaines nécessitant une optimisation comme en ingénierie biomédicale pour estimer l'intensité des distributions avec l'imagerie cérébrale par résonance magnétique [SVLM98], ou en traitement de l'image ou de la parole ([BPV<sup>+</sup>08], [GGN08]).

Pour plus de clarté,  $NP$  désigne dans la suite, comme dans l'article original, le nombre d'individus de la population. Pour bien distinguer chaque gène de chaque individu à chaque itération,  $X_k^r(i)$  représente le  $i^{\text{ème}}$  gène parmi les  $d_{ion}$  gènes du  $r^{\text{ème}}$  individu – choisi parmi les  $NP$  individus – de la  $k^{\text{ème}}$  génération.

Cette métaheuristique à population fait évoluer les individus par une recombinaison "géométrique" des vecteurs de paramètres (ou individus) et chaque individu  $X_k^r$  est comparé, à chaque itération, à un nouvel individu  $X_{k,mult}^r$  – croisant des gènes de  $X_k^r$  et  $X_{k,trial}^r$  – qui pourrait éventuellement remplacer  $X_k^r$  à la génération suivante. Ces deux caractéristiques de recombinaison et de sélection différencient l'algorithme d'évolution différentielle des algorithmes génétiques. La formation du vecteur  $X_{k,trial}^r$  servant à la *recombinaison* est appelée *différenciation*. L'Algorithme 5 décrit en détail la méthode.

Dans le cas du DE, l'ajustement des constantes de différenciation<sup>18</sup> et de recombinaison joue un rôle prépondérant dans la convergence vers une solution optimale :  $F \in [0; 2]$  est généralement

<sup>18</sup>Attention à ne pas confondre la constante de différenciation  $F$  avec la fonction de coût  $F_{\text{coût}}$ .

**Algorithme 5** : Algorithme d'évolution différentielle (DE)

**Entrées** :  $NP$  : nombre d'individus de la population,  $N_{iter}$  : nombre d'itérations,  $F_{\text{coût}}$  : fonction de coût,  $F$  : facteur de différenciation,  $CR$  : constante de recombinaison.

**Sorties** :  $x_{opt}$ , minimisant  $F_{\text{coût}}$ .

**Initialisation** :  $X$  aléatoire uniforme dans  $(D_{\text{ion}})^{NP}$ ,  $x_{opt}$ .

**pour**  $k$  de 0 à  $(N_{iter} - 1)$  **faire**

1) *Differentiation* :

le  $r^{\text{ème}}$  nouveau vecteur de paramètre potentiel,  $X_{k,trial}^r$ , est généré en additionnant  $X_k^{r_1}$ , tiré uniformément parmi les individus de la  $k^{\text{ème}}$  génération, et la différence pondérée entre deux autres individus de la population,  $X_k^{r_2}$  et  $X_k^{r_3}$ , avec  $r_1 \neq r_2 \neq r_3$ , ainsi :

$$\forall r = 1, \dots, NP, \quad X_{k,trial}^r = \min(\max(X_k^{r_1} + F \cdot (X_k^{r_2} - X_k^{r_3}), X_{min}), X_{max}), \quad (2.4)$$

2) *Recombinaison* :

Le  $r^{\text{ème}}$  individu "mutant" de la  $k^{\text{ème}}$  génération,  $X_{k,mut}^r$ , hérite des gènes de  $X_{k,trial}^r$  avec une probabilité  $CR$  propre à chaque gène, ie on génère  $u = \mathcal{U}(0, 1)$  et :

$$\begin{aligned} \forall i = 1, \dots, d_{\text{ion}}, \quad \forall r = 1, \dots, NP, \\ X_{k,mut}^r(i) = \begin{cases} X_{k,trial}^r(i) & \text{si } u < CR \\ X_k^r(i) & \text{sinon.} \end{cases} \end{aligned} \quad (2.5)$$

3) *Sélection* :

$$X_{k+1}^r = \begin{cases} X_{k,mut}^r & \text{si } F_{\text{coût}}(X_{k,mut}^r) \leq F_{\text{coût}}(X_k^r) \\ X_k^r & \text{sinon} \end{cases}$$

$x_{opt} = x \in \{X_k^1, X_k^2, \dots, X_k^{NP}\}$  t.q.  $\forall X_k^i \in \{X_k^1, X_k^2, \dots, X_k^{NP}\}, F_{\text{coût}}(x) \leq F_{\text{coût}}(X_k^i)$

retourner  $x_{opt}$

fixé à 0,5 et  $CR$  à 0,9. Ces valeurs d'hyperparamètres sont suggérées dans [SP97] et dans [FJ04] où des études expérimentales sur des benchmarks de fonctions sont réalisées. Il n'existe aucune démonstration théorique de la convergence de l'évolution différentielle en fonction de la valeur de ces deux hyperparamètres. Néanmoins, le choix de  $F$  peut s'expliquer par le fait que s'il est trop petit, la différenciation des vecteurs nous mènera vers une recherche locale autour de  $X_k^{r1}$  et, dans le cas où  $F$  est trop grand, les nouveaux vecteurs potentiels risquent de "sortir" trop souvent des contraintes imposées à l'espace de recherche. C'est donc en partie grâce au choix de  $F$  que les minima locaux sont évités. Par ailleurs,  $CR$  doit être suffisamment proche de 1 pour que de nouveaux vecteurs de paramètres entrent en lice. Nous avons également fixé  $NP = 300$  et  $N_{iter} = 300$ .

La valeur finale de la fonction de coût est de l'ordre de  $1,4 \cdot 10^{-4}$  et l'estimation prend environ 50 min. Le tableau 2.3 de la section *Bilan de comparaison* synthétise les valeurs des paramètres obtenues après estimation.

### Des variantes combinées du recuit simulé et de l'évolution différentielle

L'élaboration de variantes autour des algorithmes proposés devait nous permettre, en évitant les minima locaux, d'améliorer la rapidité (en terme de temps d'exécution) de l'estimation tout en fournissant une solution pour laquelle l'erreur serait très faible. Le test de ces nouvelles méthodes ne s'est pas avéré satisfaisant dans le sens où, d'une part, la complexification des algorithmes augmentait le temps d'exécution et d'autre part, les solutions trouvées n'étaient pas meilleures que celles obtenues avec l'évolution différentielle. Cependant, ce travail fût intéressant du point de vue conceptuel, nous le présentons donc dans cette sous-section.

La première variante que nous avons tenté d'introduire combinait, d'un côté, les mouvements dans l'espace de recherche guidés par la technique de recombinaison de l'algorithme d'évolution différentielle et de l'autre, l'exploration vaste et la diminution de l'amplitude du mouvement – au fur et à mesure que l'on s'approche d'une solution – inspirées du recuit simulé. L'algorithme de base est donc celui de l'évolution différentielle classique et la variante a été introduite au moment de la sélection de l'individu. Nous avons ainsi remplacé l'étape

$$X_{k+1}^r = \begin{cases} X_{k,mut}^r & \text{si } F_{\text{coût}}(X_{k,mut}^r) \leq F_{\text{coût}}(X_k^r) \\ X_k^r & \text{sinon} \end{cases}$$

par

$$X_{k+1}^r = \begin{cases} X_{k,mut}^r & \text{si } F_{\text{coût}}(X_{k,mut}^r) \leq F_{\text{coût}}(X_k^r) \\ \text{sinon} & \begin{cases} X_{k,mut}^r & \text{si } p \leq \exp(-\Delta f/T) \text{ où } p = \mathcal{U}(0,1) \\ X_k^r & \text{sinon} \end{cases} \end{cases}$$

en introduisant les paramètres de température,  $T_{init}$ ,  $T$ , et de gradient de températures,  $\Delta f$ , définis dans le recuit simulé (voir l'Algorithme 2 du chapitre 1). Le gradient de température est alors calculé pour chaque individu  $X_k^r$  pour lequel  $F_{\text{coût}}(X_{k,mut}^r) > F_{\text{coût}}(X_k^r)$  et l'on génère une valeur aléatoire uniforme dans  $[0, 1]$  chaque fois que cette dernière condition est réalisée. Nous augmentons donc le nombre de calculs effectués à chaque itération par rapport à la méthode d'évolution différentielle classique.

En résumé, d'un côté nous autorisons l'exploration de zones n'améliorant pas la valeur de la fonction de coût en acceptant des solutions "moins bonnes", donc nous ralentissons le processus de convergence. D'un autre côté, cette méthode de sélection peut permettre de s'éloigner de minima locaux. Or, en pratique, l'évolution différentielle ne semble pas tomber dans les minima locaux. Cette méthode présente donc, après analyse, peu d'intérêt en comparaison avec celle de

l'évolution différentielle, ce qu'ont confirmé les résultats expérimentaux<sup>19</sup>. Une méthode similaire a été présentée par K. Liu dans [LDK07] sur un benchmark de fonctions et montrait malgré tout des performances améliorées par rapport au DE classique. Cette dernière remarque justifie donc la nécessité de tester les algorithmes d'optimisation sur notre problème concret d'estimation de paramètres de modèles neuronaux à partir de données expérimentales.

Intéressons-nous maintenant à la deuxième variante. Celle-ci utilisait également le gradient de température défini dans le recuit simulé et ralentissait l'estimation par l'introduction de nouveaux vecteurs tirés aléatoirement de manière uniforme. Cependant, elle s'inspirait cette fois d'une méthode utilisée en filtrage particulaire pour évaluer la dispersion des solutions candidates dans l'espace image de  $F_{\text{coût}}$  au cours du processus d'estimation, et tenter de "concentrer" les solutions dans la zone minimisant  $F_{\text{coût}}$ . Le pseudo-code présenté dans l'algorithme 6 décrit la méthode développée. La base de cet algorithme est le DE classique. On affecte à chaque vecteur de paramètres (ou particule)  $X_k^r$  (parmi les  $NP$ ) un poids normalisé  $w_k^r$  :

$$w_k^r = \frac{e^{\left(\frac{-F_{\text{coût}}(X_k^r)}{T_k}\right)}}{\sum_{r=1}^{NP} \left(e^{\left(\frac{-F_{\text{coût}}(X_k^r)}{T_k}\right)}\right)}$$

où  $T_k$  est la température à l'instant  $k$  ; cette dernière varie de manière linéaire à chaque itération du processus du DE. Puis à chaque itération, on définit  $N_{eff}$  servant à évaluer la dispersion des solutions dans l'espace image de  $F_{\text{coût}}$  :

$$N_{eff} = \frac{1}{\sum_{r=1}^{NP} (w_k^r)^2}.$$

En effet, si toutes les particules participent, ie si elles ont toutes un poids très proche  $w_k^r \simeq 1/N$ , alors  $N_{eff} \simeq \frac{1}{\sum_{r=1}^{NP} (1/N)^2} \simeq \frac{1}{1/N} \simeq N$ . Si  $N_{eff} < \nu \times N$ , alors pour chaque  $r \in [1, NP]$ , on tire aléatoirement de manière uniforme dans  $D_{\text{ion}}$  un nouveau vecteur de paramètres  $X_{k,\text{potentiel}}^r$  qui remplacera ou non  $X_k^r$  suivant la méthode de sélection du recuit simulé. La valeur  $\nu$  est une constante que nous avons choisie égale à 0,6.

Avec cette variante, comme avec la précédente, nous augmentons le nombre d'opérations à chaque itération et complexifions donc l'algorithme sans en améliorer la rapidité de convergence. De même que dans la première variante, la modification ajoutée à l'algorithme de base permettrait d'éviter les minima locaux ; or le piège des minima locaux n'est pas la principale faiblesse de l'algorithme d'évolution différentielle. Cette variante a donc également été abandonnée. Nous reviendrons, dans la section 2.2, à des stratégies se fondant uniquement sur l'évolution différentielle ; nous les avons proposées et testées ultérieurement afin d'accélérer la convergence toujours en évitant les minima locaux.

### Les essais particuliers

Cette métaheuristique introduite en 1995 par J. Kennedy et R. Eberhart dans [KE95] s'inspire, comme son nom l'indique, de l'évolution d'un essaim d'individus, un vol d'oiseaux par exemple, duquel un comportement complexe "intelligent" – convergence vers un minimum – peut émerger avec des règles de comportement individuel simples. Les individus sont ici appelés

<sup>19</sup>Ces résultats ne sont pas présentés ici, mais montrent essentiellement un allongement important du temps de simulation et des valeurs estimées très proches des solutions trouvées avec l'évolution différentielle classique.

---

**Algorithme 6** : Deuxième variante combinée de l'évolution différentielle (recuit simulé + filtrage particulière).

---

**Entrées** :  $NP$  : nombre d'individus de la population,  $N_{iter}$  : nombre d'itérations,  $f$  : fonction de coût,  $F$  : facteur de différenciation,  $CR$  : constante de recombinaison,  $T_{init}$  : température initiale.

**Sorties** :  $x_{opt}$ , minimisant  $f$ .

**Initialisation** :  $X$  aléatoire uniforme dans  $(D_{ion})^{NP}$ ,  $x_{opt}$ .

**pour**  $k$  de 0 à  $(N_{iter} - 1)$  **faire**

1) *Differentiation* :

$X_k^{r_1}$ ,  $X_k^{r_2}$  et  $X_k^{r_3}$ , avec  $r_1 \neq r_2 \neq r_3$ , sont tirés uniformément parmi les  $NP$  individus

$$\forall r = 1, \dots, NP, \quad X_{k,trial}^r = \min(\max(X_k^{r_1} + F \cdot (X_k^{r_2} - X_k^{r_3}), X_{min}), X_{max}), \quad (2.6)$$

2) *Recombinaison* :

On génère  $u = \mathcal{U}(0,1)$  et :

$$\forall i = 1, \dots, d_{ion}, \quad \forall r = 1, \dots, NP, \quad X_{k,mut}^r(i) = \begin{cases} X_{k,trial}^r(i) & \text{si } u < CR \\ X_k^r(i) & \text{sinon.} \end{cases} \quad (2.7)$$

3) *Sélection* :

$$X_{k+1}^r = \begin{cases} X_{k,mut}^r & \text{si } F_{fit}(X_{k,mut}^r) \leq F_{fit}(X_k^r) \\ X_k^r & \text{sinon} \end{cases}$$

4) *Évaluation de la dispersion et nouvelle sélection* :

$$w_k^r = \frac{e\left(\frac{-F_{coût}(X_k^r)}{T}\right)}{\sum_{r=1}^{NP} \left(e\left(\frac{-F_{coût}(X_k^r)}{T}\right)\right)}$$

$$N_{eff} = \frac{1}{\sum_{r=1}^{NP} (w_k^r)^2}$$

**si**  $N_{eff} < 0,6 \times N$  **alors**

**pour**  $r$  de 1 à  $NP$  **faire**

$X_{k,potentiel}^r = \mathcal{U}(D_{ion})$

**si**  $(F_{coût}(X_{k,potentiel}^r) - F_{coût}(X_{k+1}^r)) < 0$  **alors**

$X_{k+1}^r = X_{k,potentiel}^r$

**sinon**

      Tirer  $p$  dans  $[0,1]$

**si**  $p \leq \exp\left(-\frac{(F_{coût}(X_{k,potentiel}^r) - F_{coût}(X_{k+1}^r))}{T}\right)$  **alors**

$X_{k+1}^r = X_{k,potentiel}^r$

$T = k \cdot T$

$x_{opt} = x \in \{X_k^1, X_k^2, \dots, X_k^{NP}\}$  t.q.  $\forall X_k^i \in \{X_k^1, X_k^2, \dots, X_k^{NP}\}, F_{coût}(x) \leq F(X_k^i)$

retourner  $x_{opt}$

---

*particules*. L'évolution de chaque particule  $i$  dépend à la fois de sa vitesse actuelle, de la meilleure solution et de la meilleure solution obtenue dans le voisinage de  $i$  ou de la meilleure solution de la génération  $t$ .  $x$  représente ici aussi un vecteur de paramètres, on note  $x(t)$  la position de ce vecteur-particule à l'instant  $t$ ,  $v(t)$  sa vitesse au même instant.  $p(t)$  désigne la meilleure position trouvée jusqu'alors sur l'ensemble des particules. La meilleure particule de la génération créée à l'instant  $t$  est appelée  $g(t)$ .  $X$  représente, comme précédemment, la matrice formée des  $N$  vecteurs particules  $x$ , ainsi le  $i^{\text{ème}}$  vecteur  $x$  de la population à l'instant  $t$  est noté  $X_t^i$ . De la même façon, on note  $V$  la matrice constituée par les vecteurs vitesses<sup>20</sup>. Par ailleurs,  $\phi_1$ ,  $\phi_2$  et  $\omega$  sont des paramètres permettant de faire varier la vitesse de la particule ; généralement, ces trois paramètres ne sont pas choisis indépendamment ([CK02][Tre03]) et  $\omega < 1$ . Plus  $\omega$  est proche de 1 plus la convergence est lente, mais plus l'exploration de l'espace de recherche est exhaustive. Le pseudo-code Algorithme 7 décrit le déroulement de cette méthode. Dans la suite,  $\mathcal{U}(0, 1)$  désigne une valeur tirée aléatoirement de manière uniforme dans l'intervalle  $[0; 1]$ .

---

**Algorithme 7** : Algorithme d'optimisation par essaims particulaires.

---

**Entrées** :  $N_{iter}$  : nombre d'itérations,  $F_{coût}$  : fonction de coût,  $N$  : nombre de particules.

**Sorties** :  $x_{opt}$ , minimisant  $F_{coût}$ .

**Initialisation** :  $X = X_0$  aléatoire uniforme dans  $D_K$ ,  $v_i(0), i = 1, 2, \dots, N$ , aléatoires uniformes,  $x_{opt} = x \in \{X_k^1, X_k^2, \dots, X_k^{NP}\}$  t.q.  $\forall X_k^i \in \{X_k^1, X_k^2, \dots, X_k^{NP}\}, F_{coût}(x) \leq X_k^i$ ,  
 $p = x_{opt}$ .

**pour**  $t$  de 0 à  $(N_{iter} - 1)$  **faire**

**pour**  $i$  de 1 à  $N$  **faire**

$$\begin{cases} V_t^i = \omega.V_t^i + \phi_1.\mathcal{U}(0, 1).(p(t) - x) + \phi_2.\mathcal{U}(0, 1).(g(t) - X_t^i) \\ X_t^i = X_t^i + V_t^i \end{cases}$$

$$\begin{cases} x_{opt} = x \in \{g(t), X_k^1, X_k^2, \dots, X_k^{NP}\} \text{ t.q. } \forall X_k^i \in \{g(t), X_k^1, X_k^2, \dots, X_k^{NP}\}, F_{coût}(x) \leq \\ F(X_k^i) \\ g(t) = x_{opt} \end{cases}$$

retourner

$$p(t) = x_{opt} = x \in \{X^1, X^2, \dots, X^{NP}\} \text{ t.q. } \forall X_k^i \in \{X^1, X^2, \dots, X^{NP}\}, F_{coût}(x) \leq F(X^i)$$


---

Les hyperparamètres utilisés pour la simulation étaient :  $\phi_1 = 0,7$  et  $\phi_2 = 0,6$ . Les résultats sont comparés dans la section suivante à ceux obtenus avec les autres méthodes.

## Bilan de comparaison entre recuit simulé, algorithmes génétiques et évolution différentielle

La première phase du travail consistait à trouver une méthode d'optimisation fournissant à la fois une erreur finale faible et une exécution rapide de la simulation en minimisant la complexité de l'algorithme. Nous avons donc rejeté les variantes combinant plusieurs méthodes en raison de leur complexité calculatoire dans la mesure où l'évolution différentielle, seule, permettait de trouver une "bonne" solution en limitant le nombre d'opérations, rendant ainsi l'estimation possible en un temps raisonnable (de l'ordre de l'heure). Il reste donc à comparer les résultats obtenus avec les algorithmes génétiques, le recuit simulé, l'évolution différentielle et les essaims particulaires.

---

<sup>20</sup>Par exemple,  $V_t^i$  est le vecteur vitesse  $v(t)$  de la  $i^{\text{ème}}$  particule  $x$  à l'instant  $t$

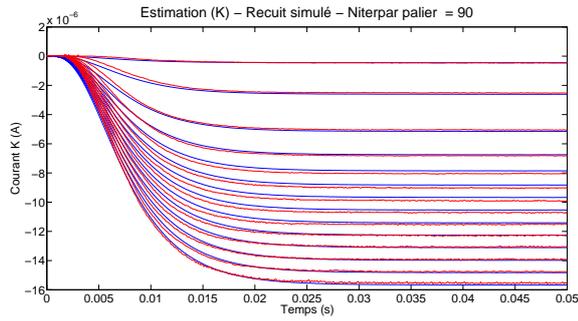
Paramètres	RS	AG	DE
$g_K (S)$	$1,69e^{-5}$	$1,33e^{-5}$	$1,66e^{-5}$
$\tau_n (s)$	$4,10e^{-3}$	$3,64e^{-3}$	$3,96e^{-3}$
$E_K (V)$	$-4,49e^{-1}$	$-6,58e^{-1}$	$-4,46e^{-1}$
$V_{offset} (V)$	$-1,73e^{-1}$	$-2,18e^{-1}$	$-1,53e^{-1}$
$V_{pente} (V)$	$5,18e^{-2}$	$9,04e^{-2}$	$4,11e^{-2}$
Erreur finale relative	$6,7 \cdot 10^{-4}$	$2,0 \cdot 10^{-3}$	$1,4 \cdot 10^{-4}$
Temps d'estimation	6j et 6h	1 h 55 min	50 min

TAB. 2.3 – Paramètres estimés du canal potassique.

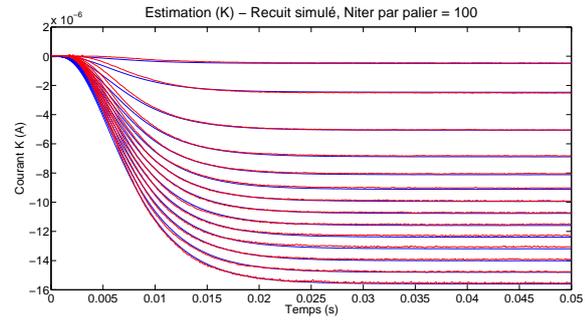
Les valeurs des paramètres estimés et le temps de l'estimation obtenus avec les algorithmes génétiques (AG), le recuit simulé (RS), l'évolution différentielle (DE) sont regroupés dans le tableau 2.3. Visuellement, les courbes obtenues après estimation des paramètres donnent l'avantage au recuit simulé (fig. 2.1b)) et à l'évolution différentielle (fig. 2.1d)). Si les courbes sont superposées pour le DE dans la partie linéaire de la réponse potassique, elles ne le sont pas tout à fait au niveau des changements de pentes durant les 15 premières millisecondes. L'étude de la section suivante permettra de déterminer si cet écart provient de la nature des données de référence ou d'une "mauvaise" estimation des paramètres. Le coût final est le plus faible avec l'évolution différentielle avec, en deuxième position, le recuit simulé et en dernière position les algorithmes génétiques. Quant au temps de réalisation de l'estimation, il est le plus réduit avec l'évolution différentielle. Bien que les valeurs des paramètres estimés avec le recuit et l'évolution différentielle paraissent très proches, les courbes de référence et les courbes simulées ne sont pas totalement superposées en régime permanent avec la méthode du recuit ; ceci indique une sensibilité importante aux valeurs de certains paramètres et prouve la nécessité de leur ajustement. La figure 2.2 représente les courbes d'évolution de la fonction de coût pour les algorithmes génétiques et l'évolution différentielle pour 300 individus et 300 itérations ; le recuit simulé n'apparaît pas sur cette courbe car il n'est pas comparable aux autres en terme de nombre d'itérations et n'est pas une métaheuristique à population. On peut constater que la valeur de la fonction de coût chute moins rapidement pour les AG que pour le DE et stagne à une valeur d'environ  $25 \cdot 10^2$  à partir de la 140<sup>ème</sup> itération. Les courbes sont données pour une réalisation représentative de chacune des optimisations avec une initialisation commune de la population.

Concernant les essais particuliers, le temps d'estimation des paramètres pour un même nombre d'individu ( $N = NP = 300$ ) et un même nombre d'itération ( $N_{iter} = 300$ ) était comparable à celui de l'évolution différentielle, voire meilleur (environ 10 minutes de moins). Par contre, l'erreur finale dépendait fortement de l'initialisation : une fois sur quatre en moyenne, l'erreur finale était comparable à celle obtenue avec l'évolution différentielle, ce qui indiquait la bonne qualité de la solution ; le reste du temps, l'erreur était très importante et la solution trouvée fournissait des courants potassiques dont l'allure est présentée figure 2.1.

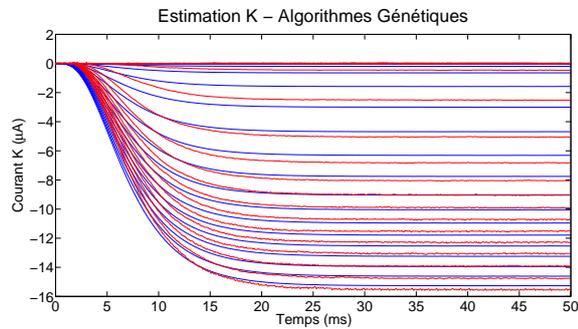
Nous avons malgré tout tenté d'estimer les paramètres du canal sodique à l'aide de notre implémentation des essais particuliers. Cette méthode ne fournit malheureusement aucun résultat probant (voir figure 2.3). Des études comparatives de différentes méthodes d'optimisation, comme dans [VT04], mettent en évidence des situations où la technique des essais particuliers peut être mise en échec lorsque le nombre de paramètres augmente et confirme le fait que la bonne convergence est souvent conditionnée par l'initialisation de l'algorithme. Des variantes ont d'ailleurs été proposées dans la littérature pour remédier à ces faiblesses [RV02] ; nous ne nous y sommes pas intéressés dans ces travaux de thèse.



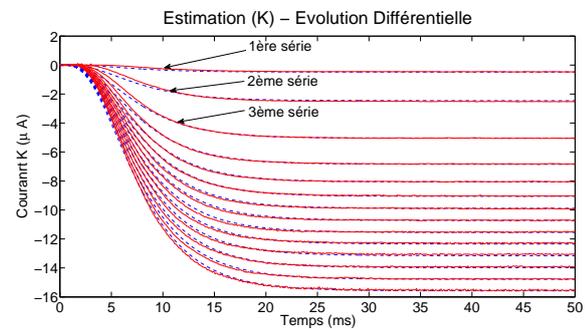
(a) Recuit : 90 itérations par palier.  $T_{init} = 1000$  et  $T_{fin} = 10^{-9}$ .



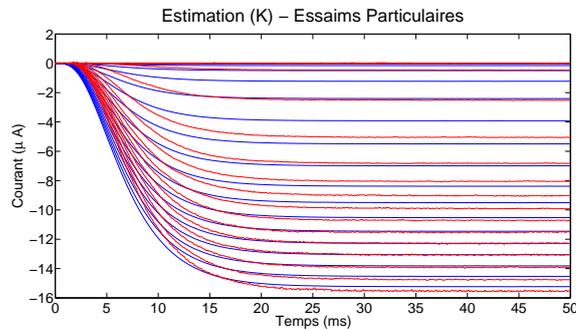
(b) Recuit : 100 itérations par palier.  $T_{init} = 1000$  et  $T_{fin} = 10^{-9}$ .



(c) Algorithmes génétiques.  $NP = 300$ ,  $N_{iter} = 300$ .



(d) Évolution différentielle.  $NP = 300$ ,  $N_{iter} = 300$ .



(e) Essais particulaire - exemple.  $N = 300$ ,  $N_{iter} = 300$

FIG. 2.1 – Réponse du canal potassique avec les paramètres estimés à l’aide du recuit simulé, des algorithmes génétiques, des essais particuliers pour un cas d’optimisation non satisfaisant et de l’évolution différentielle. En rouge le courant de référence et en bleu, le courant estimé.

Nous avons donc retenu la méthode de l’évolution différentielle classique pour l’implémen-

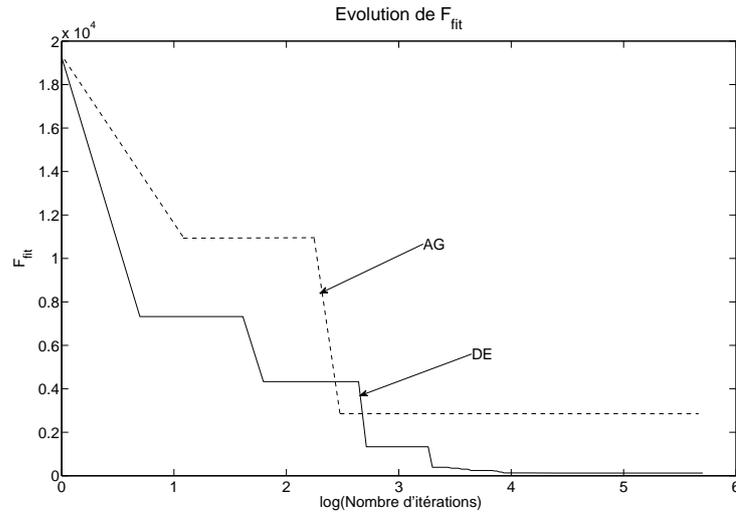
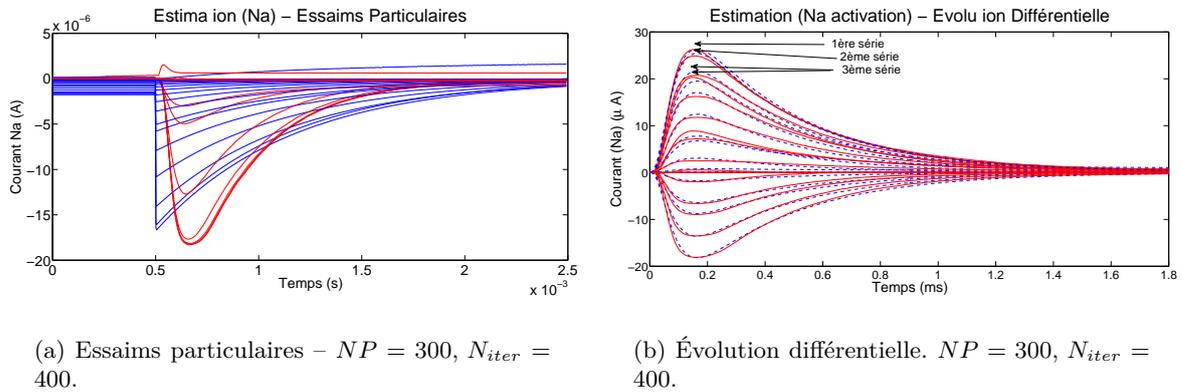


FIG. 2.2 – Évolution de la fonction de coût avec les algorithmes génétiques et l'évolution différentielle.  $NP = N = 300$ ,  $N_{iter} = 300$ .

tation de l'estimation des paramètres du modèle d'Hodgkin–Huxley à partir d'enregistrements de courants ioniques. Nous avons alors testé l'estimation des paramètres du canal sodique avec cette technique en prenant  $NP = 400$  et  $N_{iter} = 400$  pour anticiper les ressources calculatoires nécessaires à l'estimation des 8 paramètres (3 de plus que pour le potassium). Les courbes de courants apparaissent figure 2.3. Les courbes estimées ont une allure satisfaisante malgré l'écart constaté entre courbes de références et courants estimés.



(a) Essais particuliers –  $NP = 300$ ,  $N_{iter} = 400$ .

(b) Évolution différentielle.  $NP = 300$ ,  $N_{iter} = 400$ .

FIG. 2.3 – Réponse du canal sodique avec les paramètres estimés à l'aide des essais particuliers et de l'évolution différentielle. En rouge le courant de référence et en bleu, le courant estimé.

Finalement, nous avons vérifié<sup>21</sup> que l'estimation s'effectuait correctement sur le canal modulant,  $I_{mod} = g_{mod} \cdot m \cdot h \cdot (V_{mem} - E_{mod})$ . Cependant, cette vérification reste superflue car la complexité des équations de ce canal ionique est inférieure à celle du potassium pour lequel on

<sup>21</sup>Ces résultats n'apparaissent pas ici car sont similaires à ceux obtenus sur le canal potassique, en termes de vitesse de convergence et de qualité de la solution.

obtient déjà de bons résultats d'estimation.

### 2.1.4 Validation numérique de la méthode du DE

Afin de déterminer si l'écart entre les courbes estimées et les courbes de référence (figures 2.3 et 2.4) était dû aux paramètres estimés ou bien à l'allure des courbes de références (courants issus du circuit neuromimétique) qui ne suivraient pas exactement les équations du modèle, nous avons testé l'algorithme en prenant comme courbes de références des données logicielles calculées à partir des équations du modèle sur les canaux potassique et sodique. Ceci nous a également permis d'évaluer la qualité de la solution en connaissant exactement les paramètres à estimer. Notons que nous aurions pu débiter ces travaux de thèse par cette étude théorique avant de nous focaliser sur l'applications à des données expérimentales. Cependant, le fait que l'algorithme fournisse de bons résultats sur des données non bruitées ne nous aurait pas nécessairement assuré une bonne convergence sur des données expérimentales perturbées par du bruit.

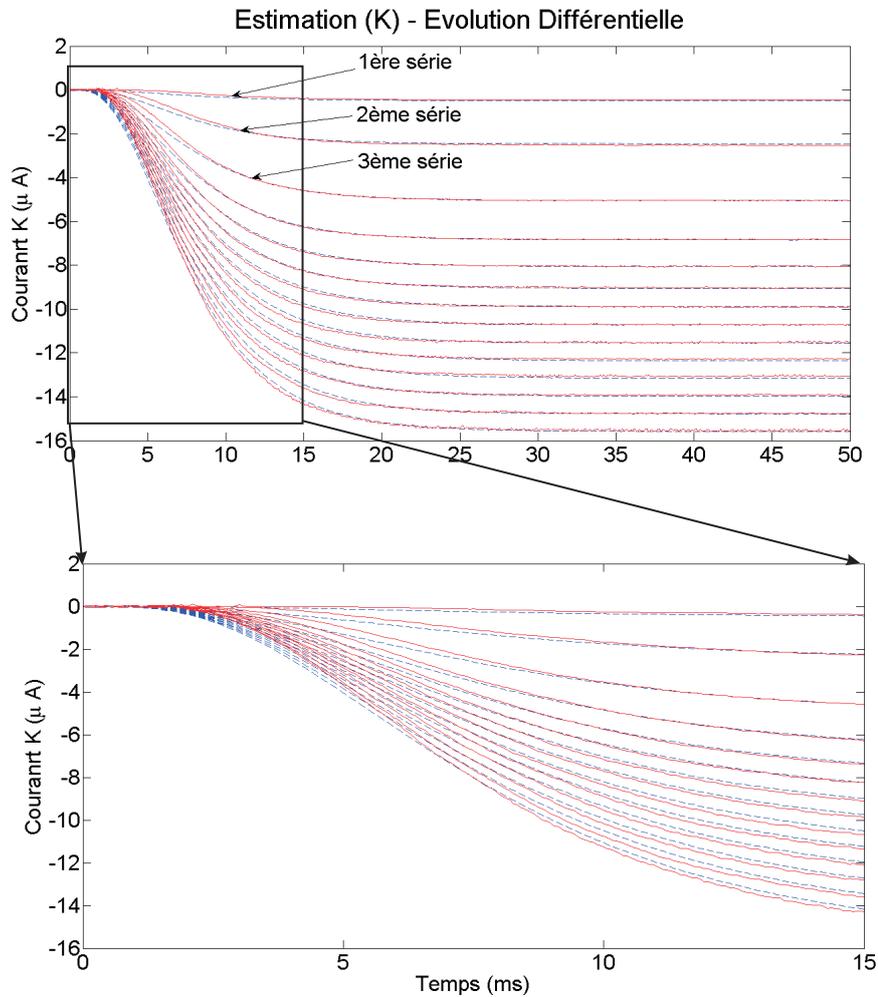


FIG. 2.4 – Zoom sur la réponse du canal potassique durant les 15 premières millisecondes.

Paramètres	Potassium			Sodium		
	Référence	Estimés		Référence	Estimés	
		25 itér.	50 itér.		100 itér.	300 itér.
$g_{\text{ion}} (S)$	$22,00e^{-4}$	$2,198e^{-4}$	$21,9998e^{-4}$	$1,711e^{-5}$	$1.290e^{-4}$	$1,7113e^{-5}$
$\tau_{\text{act}} (s)$	$3,00e^{-3}$	$2,998e^{-3}$	$3,000e^{-3}$	$5,073e^{-5}$	$4.148e^{-5}$	$5,0731e^{-5}$
$\tau_{\text{inact}} (s)$	-	-	-	$3,732e^{-4}$	$5,573e^{-4}$	$3,7321e^{-4}$
$E_{\text{ion}} (V)$	$-1,00e^{-1}$	$-1,479e^{-1}$	$-1,005e^{-1}$	$2,547e^{-1}$	$2,735e^{-1}$	$2,5473e^{-1}$
$V_{\text{offset\_act}} (V)$	$-3,7e^{-2}$	0,000	$-3,374e^{-2}$	$-1,743e^{-1}$	$-1,672e^{-1}$	$-1,7432e^{-1}$
$V_{\text{offset\_inact}} (V)$	-	-	-	$-2,251e^{-1}$	$-2.011e^{-1}$	$-2,2520e^{-1}$
$V_{\text{pente\_act}} (V)$	$1,138e^{-2}$	$2,508e^{-2}$	$1,000e^{-2}$	$3,737e^{-2}$	$4.234e^{-2}$	$3,7367e^{-2}$
$V_{\text{pente\_inact}} (V)$	-	-	-	$2,397e^{-2}$	$1,602e^{-2}$	$2,3973e^{-2}$

TAB. 2.4 – Paramètres de référence utilisés pour la validation logicielle de la méthode d'évolution différentielle et paramètres estimés après 10 et 25 itérations pour le potassium et après 100 et 300 itérations pour le sodium. Valeurs électroniques (paramètres pour le circuit neuromimétique).

## Résultats

La série de courbes de référence a été constituée en “injectant” dans les équations du modèle les paramètres suivants (tableau 2.4). Pour la caractérisation du canal potassique, nous avons appliqué 32 de sauts de tensions différents, chaque série comportant 500 points. Pour la caractérisation du canal sodique,  $2 \times 16$  séries de 700 points chacune ont servi à calculer les courants de référence pour estimer les paramètres. Les paramètres de l'algorithme d'évolution différentielle étaient les suivants :  $NP = 300$ ,  $F = 0,5$ ,  $CR = 0,9$ .

La figure 2.5 fournit l'évolution de l'erreur relative pour les deux canaux ioniques, alors que les figures 2.6, 2.7 et 2.8 représentent la réponse des courants ioniques obtenue avec les paramètres estimés à différents moments de l'optimisation (après 25 et 50 itérations pour le potassium et après 100 et 300 itérations pour le sodium). Les paramètres estimés correspondants figurent dans le tableau 2.2.

Après 45 itérations pour le potassium, les valeurs estimées des paramètres valent 99,9% des valeurs de référence ; pour le sodium, nous obtenons ce résultat après 130 itérations environ. Dans ces conditions, les courbes estimées et de référence sont totalement superposées.

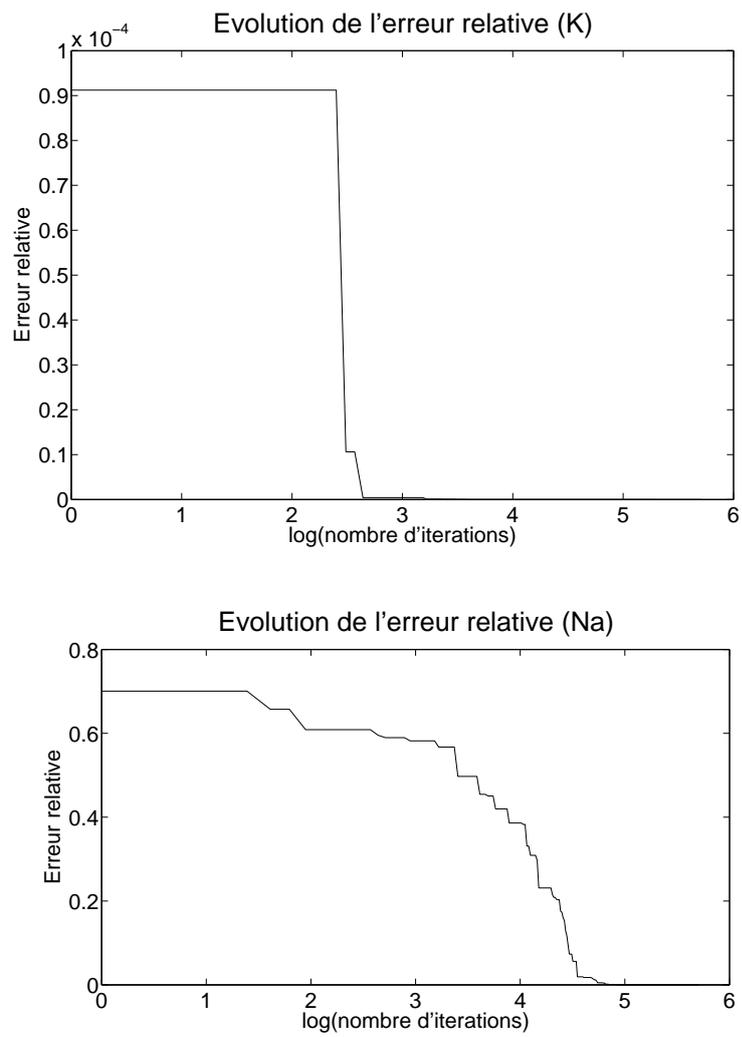


FIG. 2.5 – Evolution de l'erreur relative pour le potassium (en haut) et le sodium (en bas).

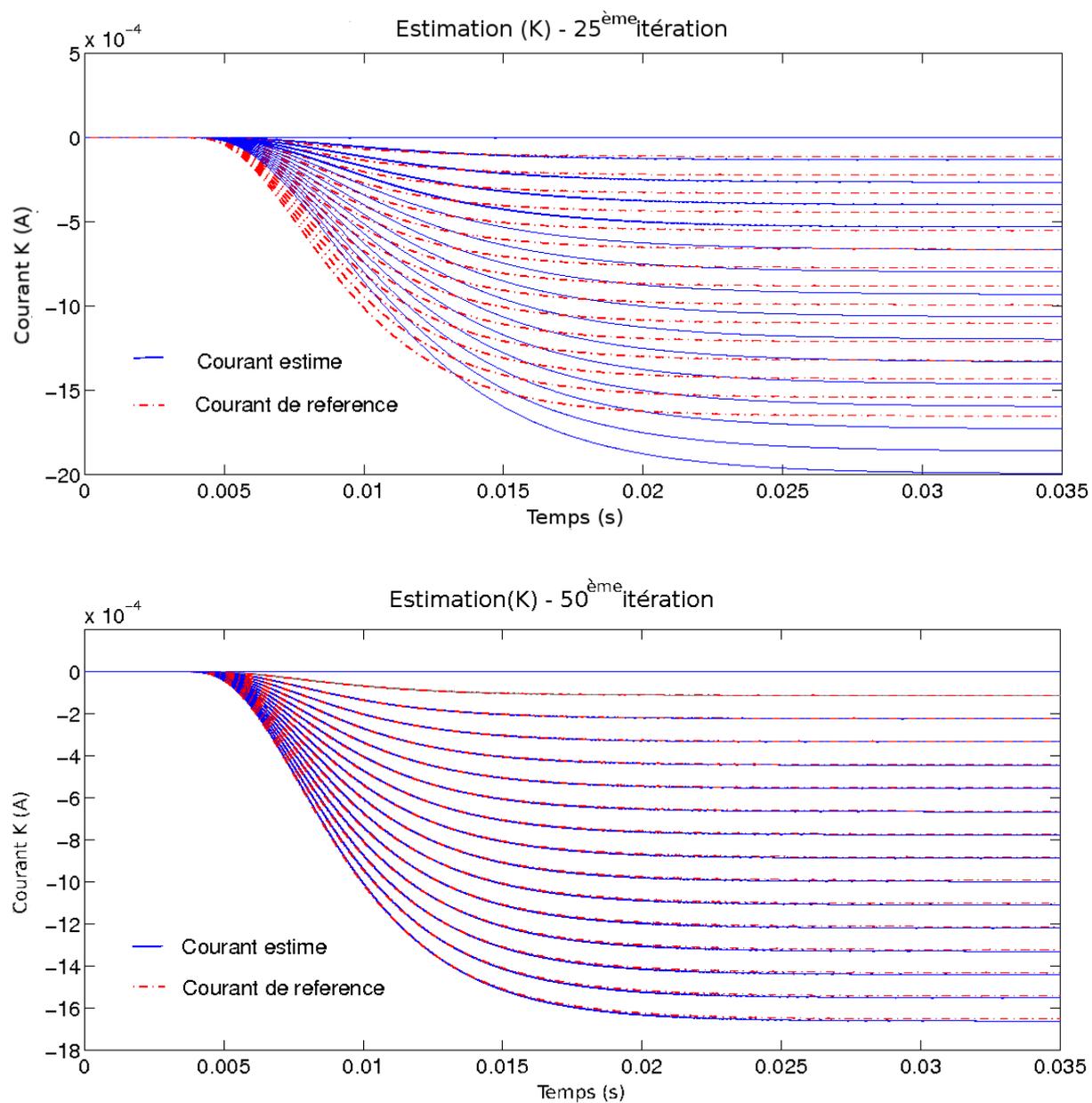


FIG. 2.6 – Réponse du canal potassique. Haut : courbes après 25 itérations. Bas : courbes après 50 itérations.

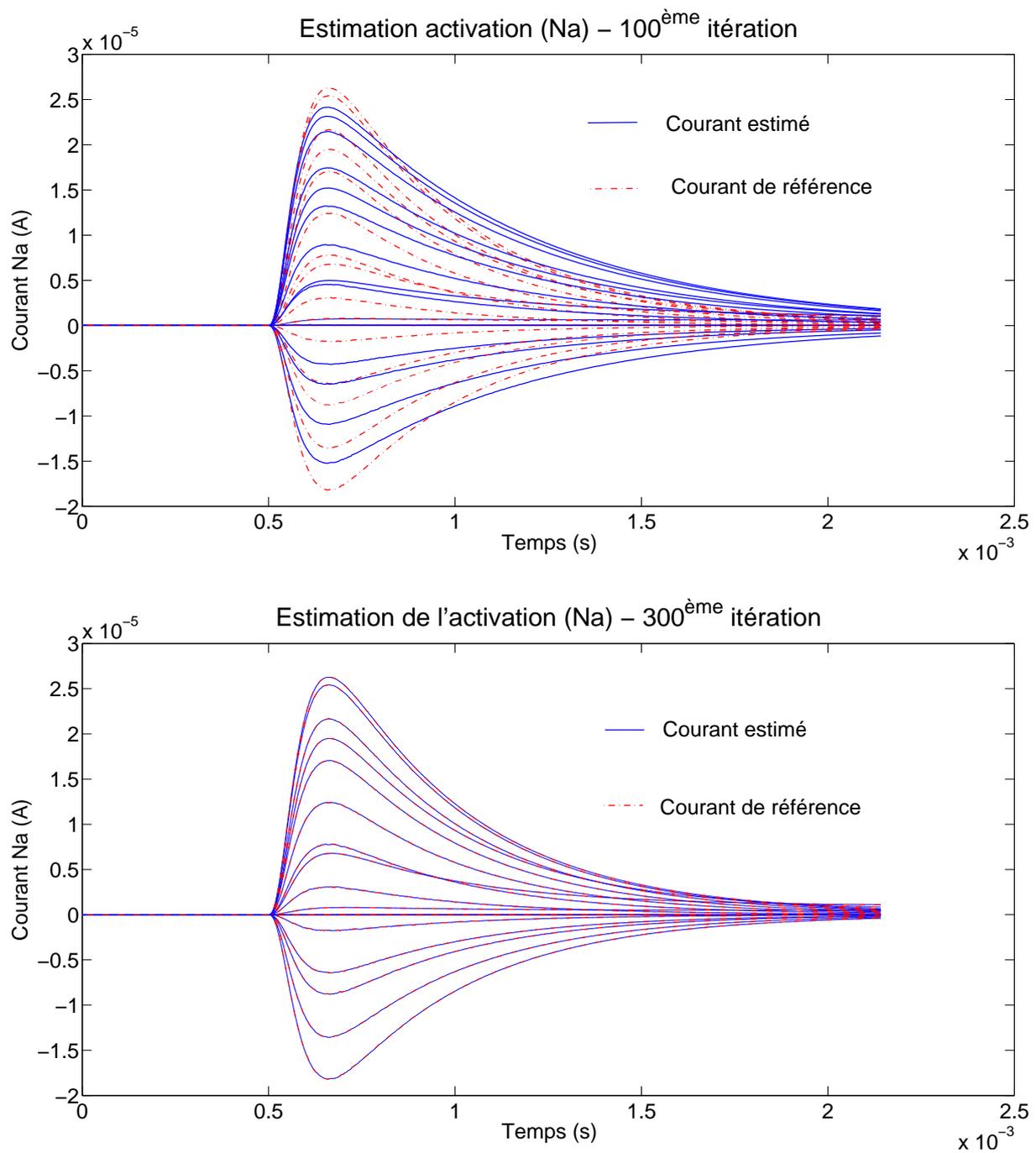


FIG. 2.7 – Réponse du canal sodique à la série de stimulations relative à la caractérisation du terme d'activation. Haut : courbes après 100 itérations. Bas : courbes après 300 itérations.

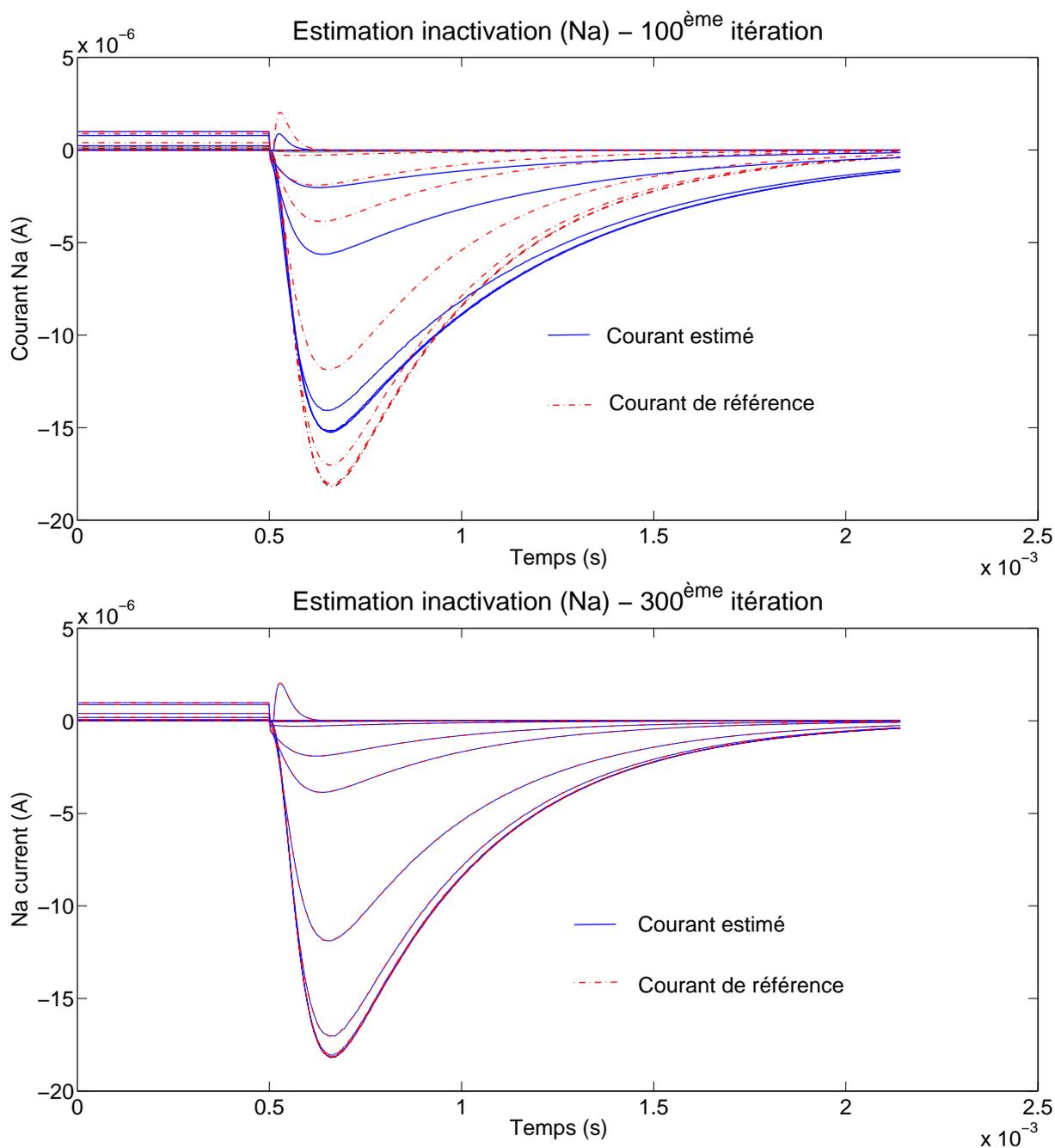


FIG. 2.8 – Réponse du canal sodique à la série de stimulations relative à la caractérisation du terme d'inactivation. Haut : courbes après 100 itérations. Bas : courbes après 300 itérations.

### Interprétation des résultats

Lorsque nous utilisons comme données de référence des courants calculés logiquement, les courbes sont totalement superposées et les paramètres sont estimés avec une précision largement supérieure au bruit généré par un système physique. Nous en déduisons donc que la disparité

entre courbes estimées et de référence aux figures 2.4 et 2.3 est due à la nature des données de référence utilisées dans la phase de comparaison des différentes méthodes : ces données de références ne “suivent” pas exactement les équations du modèle. Nous expliquerons dans le chapitre suivant l’origine de ce phénomène lié à la conception du circuit neuromimétique. L’algorithme du DE est donc bien adapté à notre problème d’optimisation, c’est pourquoi nous nous focaliserons sur cette méthode dans la section suivante en tentant d’améliorer encore ses performances.

Nous constatons également que, pour un même nombre d’individus, l’estimation des paramètres du canal sodique par le DE nécessite environ 3 fois plus d’itérations que pour celle du canal potassique. Nous avons donc tenu compte de cette observation pour les applications présentées dans le chapitre suivant.

## 2.2 Amélioration de l’algorithme d’Évolution Différentielle

Après avoir choisi la méthode de l’évolution différentielle et avoir validé son comportement avec des données de référence expérimentales et artificielles calculées numériquement, nous avons cherché à accélérer sa vitesse de convergence. D’autres stratégies de différenciation avaient déjà été développées, notamment dans [FJ04] ; les deux principales stratégies utilisent le meilleur individu,  $X_k^{\min}$  minimisant la fonction de coût, de la génération courante pour créer un nouvel individu potentiel. Dans la première stratégie, l’opération classique de différenciation est remplacée par :

$$\forall r = 1, \dots, NP, \quad X_{k,trial}^r = X_k^{\min} + F.(X_k^{r2} - X_k^{r3}) \quad (2.8)$$

où  $X_k^{r2}$  et  $X_k^{r3}$  sont tirés aléatoirement de manière uniforme parmi les individus de la population courante. Dans cette stratégie, la différenciation s’apparente à une recherche chaotique locale autour de la meilleure solution<sup>22</sup>. Dans la suite, nous noterons cette stratégie *RAND/BEST*.

Dans la seconde stratégie, au lieu de sommer trois vecteurs, quatre vecteurs sont utilisés et l’opération de différenciation est remplacée par :

$$\forall r = 1, \dots, NP, \quad X_{k,trial}^r = X_k^{r1} + F.(X_k^{r1} - X_k^{\min}) + F.(X_k^{r2} - X_k^{r3}) \quad (2.9)$$

où  $X_k^{r1}$ ,  $X_k^{r2}$  et  $X_k^{r3}$  sont également des vecteurs choisis aléatoirement parmi les individus de la population courante. La direction de recherche est alors  $X_k^{r1} - X_k^{\min} + X_k^{r2} - X_k^{r3}$ . Afin de la différencier de la précédente, cette stratégie sera appelée *RAND/MIN*.

Bien que n’étant pas plus coûteuses en temps de calcul que la stratégie classique et permettant dans certains cas de trouver rapidement la solution optimale, de telles stratégies utilisant le meilleur individu peuvent converger vers des minima locaux sans jamais explorer la région du minimum global. Nous avons donc proposé des variantes que nous présentons ci-dessous afin d’améliorer la rapidité de convergence par rapport au DE classique tout en évitant les minima locaux.

### 2.2.1 Introduction de nouvelles variantes de l’évolution différentielle

Afin de limiter le risque de trouver un minimum local, nous proposons, au lieu de considérer uniquement le meilleur individu de chaque génération, de favoriser les individus dont la valeur de la fonction de coût est faible. Pour cela, au lieu de tirer uniformément les individus servant

<sup>22</sup>Voir [FJ04] : « it looks like a chaotical local search around the current best solution »

à la différenciation, nous modifions la probabilité de tirage de telle sorte que, plus la valeur de la fonction de coût associée à un individu est faible, plus l'individu a de chance d'être tiré au sort. De cette façon, nous maintenons une certaine diversité au sein de la population tout en favorisant les régions de moindre coût.

Soit  $p_k^{r_i}$  la probabilité pour que l'individu  $X_k^{r_i}$  soit sélectionné.  $p_k^{r_i}$  est ainsi définie :

$$p_k^{r_i} = K \cdot \exp(-F_{\text{coût}}(X_k^{r_i})), \quad (2.10)$$

où  $K$  est une constante de normalisation.

Cependant, si la fonction de coût prend des valeurs élevées, toutes les probabilités  $\{p_k^{r_i}\}_{0 \leq i \leq NP}$  se retrouvent "écrasées" par la fonction exponentielle et ont toutes des valeurs similaires. Cette approche ne permet donc pas de discriminer suffisamment bien les différentes valeurs de la fonction de coût. Pour contourner cet obstacle, nous avons redéfini la probabilité associée à chaque individu  $X_k^r$  de la façon suivante :

$$p_k^r = K' \cdot \exp\left(\frac{-\kappa \cdot (F_{\text{coût}}(X_k^r) - F_{\text{coût}}(X_k^{\min}))}{F_{\text{coût}}(X_k^{\max}) - F_{\text{coût}}(X_k^{\min})}\right). \quad (2.11)$$

où  $K'$  est la constante de renormalisation,  $\kappa$  un facteur ajustable et  $X_k^{\max}$  est l'individu de la  $k^{\text{ème}}$  génération pour lequel la valeur de la fonction de coût est maximale. Cette formule mérite quelques commentaires :

- 1) En soustrayant  $F_{\text{coût}}(X_k^{\min})$  et en divisant par  $F_{\text{coût}}(X_k^{\max}) - F_{\text{coût}}(X_k^{\min})$ , nous nous assurons que l'exponentielle prendra ses valeurs dans l'intervalle  $[-1, 0]$  quel que soit l'individu, conduisant alors à des probabilités régulièrement réparties dans l'intervalle  $[0, 1]$ . Pour illustrer l'intérêt de diviser par la valeur maximale de  $F_{\text{coût}}(X_k^r)$ , la figure 2.9 représente deux courbes montrant les valeurs des probabilités obtenues selon l'équation (2.10) (ligne continue) et suivant l'équation (2.11) (en pointillés). Nous avons, pour réaliser le tracé, considéré des valeurs de  $F_{\text{fit}}$  uniformément distribuées dans  $[0; 500]$ .

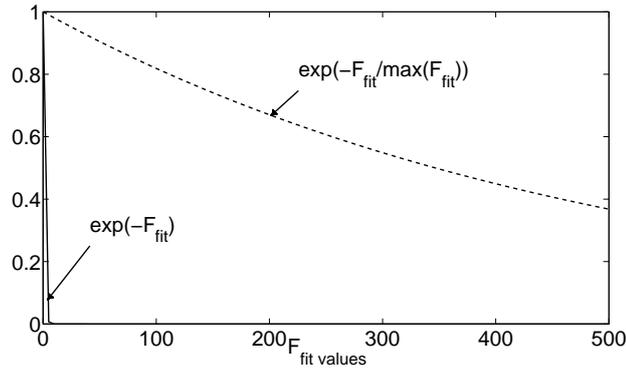


FIG. 2.9 – Justification du choix de  $p_k^{r_i}$ .

- 2) Le facteur  $\kappa$  peut être ajusté manuellement par l'expérimentateur. Pour de petites valeurs de  $\kappa$ , la distribution multinomiale constituée par l'ensemble des probabilités ainsi définies tend vers une distribution uniforme. Au contraire, si nous considérons de trop fortes valeurs de  $\kappa$ , la distribution tend vers une mesure de Dirac centrée en  $X_k^{\min}$  ; dans ce cas, presque toutes les valeurs de  $F_{\text{fit}}$  conduisent à une même probabilité proche de zéro et seul le meilleur individu à des chances d'être sélectionné.

3) Notons finalement que cette définition présente des similarités avec la probabilité d'acceptation d'une solution potentielle dans l'algorithme de recuit simulé.

Cette approche doit donc permettre d'accélérer la convergence du DE en présence de minima locaux en autorisant, contrairement aux stratégies *RAND/MIN* et *RAND/BEST*, de s'en échapper grâce au tirage de la direction de recherche déterminée par  $X_k^{r_1}$ ,  $X_k^{r_2}$  et  $X_k^{r_3}$ . Notre stratégie peut être vue comme une exploration en parallèle des régions les plus susceptibles de contenir le minimum global de notre problème d'optimisation. Néanmoins, en l'absence de minimum local, cette stratégie peut parfois nécessiter davantage d'itérations que les deux stratégies présentées précédemment.

Dans la section suivante, nous comparons 6 des stratégies du DE : la version "classique", *RAND/BEST*, *MIN/BEST* et 3 variantes sur la base de la technique introduite ci-dessus :

1) la variante du DE classique consiste à tirer les individus  $X_k^{r_i}$ , for  $i = 1, 2, 3$ , de l'équation (2.8) suivant la loi multinomiale définie précédemment ; 2) dans la variante de *RAND/BEST*, nous remplaçons le  $X_k^{\min}$  de l'équation (2.8) par un individu choisi selon la loi multinomiale, les deux autres individus étant tirés de manière uniforme ; 3) finalement, dans la variante du *RAND/MIN*, nous remplaçons le  $X_k^{\min}$  de l'équation (2.9) par un individu choisi selon la loi multinomiale, les trois autres étant tirés de manière uniforme.

## 2.2.2 Étude comparative sur un benchmark de fonctions

Afin d'évaluer la pertinence de nos variantes, nous avons comparé les différentes stratégies du DE et les variantes associées sur un ensemble de fonctions-tests appartenant au benchmark dit de De Jong [Jon75], également utilisé dans [FJ04] :

$$f_1(x) = \sum_{i=1}^3 x_i^2, -5.12 \leq x_i \leq 5.12 \quad (2.12)$$

$$f_2(x) = 100(x_1 - x_2)^2 + (1 - x_1)^2, -2.048 \leq x_i \leq 2.048 \quad (2.13)$$

$$f_3(x) = \sum_{i=1}^5 [x_i], -5.12 \leq x_i \leq 5.12 \quad (2.14)$$

$$f_4(x) = \sum_{i=1}^{10} (i \cdot x_i^4 + \varepsilon_i), -5.12 \leq x_i \leq 5.12 \quad (2.15)$$

où  $\varepsilon_i$  est un bruit blanc gaussien

$$f_5(x) = \frac{1}{0.002 + \sum_{j=1}^2 1/(c_j + \sum_{i=1}^2 (x_i - a_{ij})^6)} - \gamma, -65.536 \leq x_i \leq 65.536 \quad (2.16)$$

où  $[x]$  désigne la partie entière de  $x$ . Les paramètres sont définis ainsi :  $\gamma = 500$ ,  $c_j = 1 + j$ ,  $a_{1j} = -16(\text{mod}(16, 5) - 2)$  et  $a_{2j} = -16(\lfloor i/5 \rfloor - 2)$  où  $\text{mod}(x, y)$  représente  $x$  modulo  $y$ . On remarquera que  $f_1$  et  $f_2$  ne possèdent qu'un minimum global, tandis que  $f_3$ ,  $f_4$  et  $f_5$  ont plusieurs minima locaux. De plus,  $f_1$  est séparable, c'est-à-dire qu'elle peut être réécrite comme somme de fonctions d'une variable. Dans cette fonction, les différentes composantes de la variable  $x$  ne sont pas très dépendantes les une des autres.  $f_2$  n'est pas séparable, elle est donc plus difficile à optimiser car la "meilleure" direction de recherche dépend d'au moins deux des composantes de

$x$ .  $f_4$ , qui n'est pas séparable, est perturbée par un bruit blanc gaussien. Enfin,  $f_5$ , qui possède de nombreux minima locaux, n'est pas séparable.

Les résultats de simulations présentés plus loin correspondent aux valeurs moyennes obtenues sur 50 estimations pour chacun des algorithmes du DE testés. Toutes les variantes ont été initialisées de la même manière de sorte que la comparaison soit la plus pertinente possible. La première génération est distribuée uniformément aux sein des intervalles de définition de chaque fonction-test et le paramètre  $\kappa$  a été fixé égal à 5. Comme tous les algorithmes conduisaient approximativement à la même valeur finale des paramètres à estimer, nous ne présentons pas les résultats.

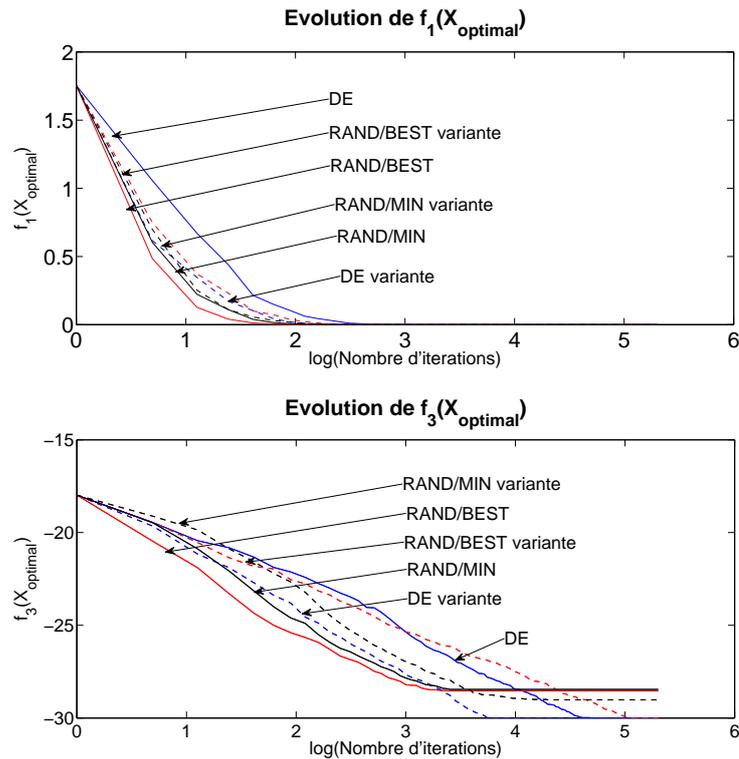


FIG. 2.10 – Évolution de  $f_1$  et  $f_3$  versus logarithme népérien du nombre d'itérations. Lignes continue : stratégies classiques du DE, lignes pointillées : variantes associées.

Afin d'illustrer le comportement des différents algorithmes, nous présentons à la figure 2.10 l'évolution moyenne de la fonction de coût pour les fonctions  $f_1$  et  $f_3$ .

Les fonctions  $f_1$  et  $f_2$  ayant uniquement un minimum global, les algorithmes d'optimisation n'ont aucune chance de tomber dans un minimum local. Il s'en suit – comme nous l'attendions – l'obtention de meilleurs résultats avec les stratégies standard *RAND/MIN* et *RAND/BEST* qu'avec le DE classique et les variantes proposées. On peut noter cependant que nos variantes convergent plus vite que le DE classique car, au lieu d'explorer aveuglément tout l'espace des solutions, elles favorisent les régions où se trouvent les meilleurs individus de chaque génération. Pour conforter cette constatation, la figure 2.11 représente les nuages de points à différents stades de l'estimation pour le DE classique et sa variante<sup>23</sup>. Après 20 itérations, on peut observer

<sup>23</sup>Sur la dernière figure représentant le nuage de point à la 90<sup>ème</sup> itération, les points sont trop proches les uns

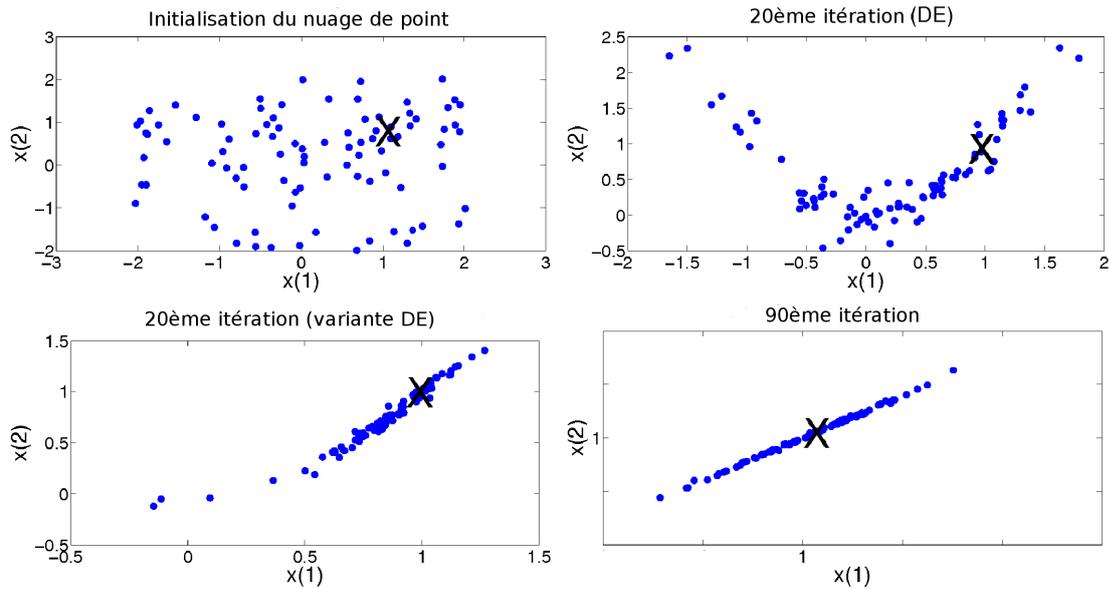


FIG. 2.11 – Évolution du nuage de points solutions pour  $f_2$  avec le DE standard et sa variante. L’initialisation est la même pour les deux algorithmes; la distribution finale est à peu près identique dans chacun des cas. Seule la distribution finale pour le DE classique apparaît. La croix  $\times$  représente le point optimal.

qu’avec la variante, les vecteurs candidats sont très proches les uns des autres, mais également très proches de l’optimum de  $f_2$ . À l’inverse, lorsqu’il s’agit du DE classique, les vecteurs de variables sont plus dispersés.

Quant aux fonctions comme  $f_3$  comportant plusieurs minima locaux, les stratégies usuelles *RAND/MIN* et *RAND/BEST* ne convergent pas vers la solution optimale. Elles sont en effet “piégées” dans des minima locaux, alors qu’au contraire, le DE classique, autant que sa variante et celle de *RAND/BEST* atteignent le minimum global. On peut remarquer de plus que l’estimation est beaucoup plus rapide avec la variante du DE classique qu’avec le DE lui-même.

Afin de tester la robustesse des approches proposées, nous avons également évalué leur vitesse de convergence pour toutes les fonctions du benchmark de De Jong. Le tableau 2.5 récapitule le nombre d’itérations nécessaires pour que  $\{(f_i(X_k^{\min}) - \min(f_i)) / (f_i(X_1^{\min}) - \min(f_i))\}_{i=1,2,3,4}$  atteigne la valeur  $10^{-4}$ , où  $\min(f_i)$  est le minimum de  $f_i$  et  $f_i(X_1^{\min})$  est la valeur minimale de la fonction de coût à la 1<sup>ère</sup> génération. On peut voir au tableau 2.5 que nos variantes convergent en moyenne plus rapidement que le DE standard. Pour les fonctions ne comportant qu’un minimum global comme  $f_1$  et  $f_2$ , les variantes sont plus lentes que les stratégies *RAND/BEST* et *RAND/MIN*. Cependant, lorsque les fonctions ont des optima locaux, les stratégies classiques ont tendance à fournir une solution locale (voir fig 2.10) sans trouver l’optimum global. Par exemple, pour la fonction  $f_3$ , *RAND/BEST* et *RAND/MIN* ne convergent respectivement que 6 fois sur 50 et 5 fois sur 50 simulations. À l’inverse, nos variantes trouvent le minimum global.

Pour conclure, sur le benchmark de fonctions, nos variantes convergent plus rapidement que le DE classique et sont plus robustes quant aux minima locaux que les stratégies *RAND/MIN* et

---

des autres pour que le logiciel puisse afficher suffisamment de chiffres significatifs. C’est pourquoi seule la valeur 1 apparaît.

	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$
<i>DE</i>	24	25	82	36	43
<i>DE</i> <sub>variante</sub>	15	19	35	19	25
<i>RAND/MIN</i>	12	13	—	9	—
<i>RAND/MIN</i> <sub>variante</sub>	14	17	—	11	32
<i>RAND/BEST</i>	9	9	—	9	—
<i>RAND/BEST</i> <sub>variante</sub>	17	20	123	22	71

TAB. 2.5 – Nombre d'itérations nécessaires pour atteindre la valeur  $10^{-4}$ . Valeur moyenne sur 50 simulations. “—” signifie que la stratégie ne converge pas.

*RAND/BEST*. En effet, en présence de minima locaux, *RAND/BEST* et *RAND/MIN* peuvent ne jamais atteindre la solution optimale en raison de la technique d'exploration centrée sur le meilleur individu de chaque génération, cet individu pouvant être éloigné, dans l'espace de recherche, de la solution optimale. Au contraire, les variantes proposées sont plus efficaces car elles recherchent en parallèle dans différentes régions “intéressantes” de l'espace des solutions. Dans la section suivante, nous appliquons ces algorithmes au modèle d'Hodgkin–Huxley afin de les comparer sur un cas réel.

### 2.2.3 Application des variantes au canal potassique du modèle d'Hodgkin–Huxley

Les données ayant servi à la comparaison des différentes stratégies et variantes associées sont les enregistrements issus du circuit neuromimétique utilisés à la section 2.1, de même que la fonction de coût. Le nombre  $NP$  d'individus est de 300 et les hyperparamètres  $F$  et  $CR$  sont inchangés. Seulement trois estimations pour chacune des stratégies testées ont été réalisées en raison de la durée des simulations.

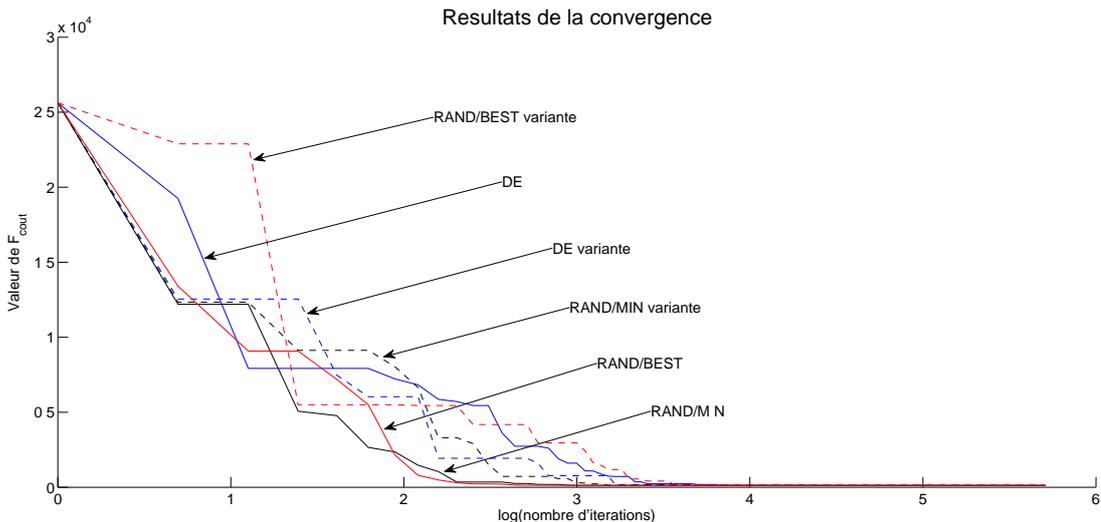


FIG. 2.12 – Évolution de la fonction de coût avec le DE standard et ses variantes. Le tracé pointillé représente l'évolution de la variante correspondant à l'algorithme standard de même couleur.

La figure 2.12 présente l'évolution de la fonction de coût pour les stratégies envisagées ainsi que leurs variantes : *RAND/MIN* converge le plus rapidement. La variante de *RAND/MIN* converge plus rapidement que le DE classique, mais *RAND/MIN* nécessite moins d'itérations que notre variante. Cependant, les trois variantes décroissent toute plus vite que la version classique du DE vers une valeur acceptable de la fonction de coût. L'explication la plus probable est qu'au cours des trois estimations, ni *RAND/MIN*, ni *RAND/BEST* ne sont "tombés" dans les quelques minima locaux de la fonction décrivant le courant potassique<sup>24</sup>.

En conclusion, nous avons développé une alternative à la technique disjointe d'estimation du voltage-clamp. La comparaison des différentes métaheuristiques nous a permis de retenir l'algorithme d'évolution différentielle pour sa rapidité de convergence, son temps d'exécution limité, sa faible complexité calculatoire et la qualité des solutions fournies. Par ailleurs, bien que complexifiant les calculs à exécuter, les variantes du DE introduites permettent d'améliorer la rapidité de convergence sans être victime des minima locaux. L'évolution différentielle et ses variantes sont donc éligibles aux applications futures. Dans le chapitre qui suit, nous avons finalement choisi d'employer la variante de la stratégie classique du DE.

---

<sup>24</sup>Les résultats auraient peut-être été davantage probants sur le canal sodique, plus complexe.

## 2.3 Résumé

Ce chapitre établit un premier contact concret avec l'optimisation numérique à travers l'implémentation et la comparaison de différents algorithmes visant à extraire les paramètres du modèle. La comparaison de plusieurs métaheuristiques (recuit simulé, algorithmes génétiques, évolution différentielle et essaims particulaires) sur des données de référence issues de mesures expérimentales nous a conduit à la sélection de l'algorithme d'évolution différentielle dont les performances dépassaient largement – en terme de qualité de la solution, rapidité de convergence (nombre d'itérations), de rapidité d'exécution (temps de calcul) et de complexité calculatoire – celles des autres algorithmes. Après avoir retenu cette méthode pour les applications futures, nous avons étudié plus en détail sa convergence en prenant, comme données de référence, des données calculées logiquement à partir des équations du modèle d'Hodgkin–Huxley. Enfin, nous avons introduit des variantes de quelques stratégies de l'évolution différentielle afin d'en améliorer la vitesse de convergence tout en évitant les minima locaux. Ces variantes ont été testées, en premier lieu, sur un jeu de fonctions-tests (benchmark de De Jong) avant d'être comparées sur l'estimation des paramètres du canal potassique dans le modèle d'Hodgkin–Huxley. Les méthodes d'extraction du modèle d'Hodgkin–Huxley explorées ici constituent donc une alternative à la technique disjointe d'estimation du voltage-clamp.



## Chapitre 3

# Applications : modélisation à partir de données biologiques et réglage automatique de circuits neuromimétiques

### Sommaire

---

<b>3.1</b>	<b>Application à des données biologiques . . . . .</b>	<b>68</b>
3.1.1	Méthode . . . . .	68
3.1.2	Résultats . . . . .	69
<b>3.2</b>	<b>Application au réglage de circuits neuromimétiques . . . . .</b>	<b>70</b>
3.2.1	Méthode . . . . .	70
3.2.2	Résultats – Discussion . . . . .	75
<b>3.3</b>	<b>Estimation des paramètres d’un modèle à partir de la tension de membrane . . . . .</b>	<b>78</b>
3.3.1	Méthode . . . . .	78
3.3.2	Quelques résultats . . . . .	80
<b>3.4</b>	<b>Résumé . . . . .</b>	<b>83</b>

---

« Un expert, c’est quelqu’un qui a fait toutes ses erreurs dans un champ réduit d’applications. »  
(Niels Bohr)

Le chapitre précédent présentait la comparaison de différentes méthodes d'optimisation qui nous a permis de sélectionner la plus appropriée à notre problème de modélisation. La méthode retenue a alors été validée théoriquement sur des données issues d'un circuit neuromimétique et des données calculées numériquement. Nous avons ainsi pu nous assurer de sa robustesse face au bruit et de sa convergence face au modèle d'Hodgkin-Huxley. Nous abordons donc maintenant deux applications auxquelles a donné lieu l'élaboration de notre méthode d'estimation de paramètres de modèles neuronaux, alternative à celle du voltage-clamp : l'estimation des paramètres du même modèle à partir d'enregistrements biologiques de courants sodiques, puis l'application au réglage de circuits neuromimétiques. Cependant, l'enregistrement des courants ioniques d'un neurone biologique demeure difficile pour les neurophysiologistes ; l'idéal serait donc de pouvoir estimer les paramètres d'un modèle de neurone directement à partir d'enregistrements de sa tension de membrane, la technique expérimentale étant beaucoup plus aisée. Une partie du travail de thèse, présentée en troisième section de ce chapitre, a donc consisté en l'élaboration d'une méthode répondant à ce critère. Cette approche oblige à prendre en compte le comportement dynamique du neurone.

### 3.1 Application à des données biologiques

Nous proposons dans cette section d'estimer les paramètres du canal sodique dans le modèle d'Hodgkin-Huxley à constantes de temps fixes à partir de données biologiques. Les paramètres à estimer sont donc les mêmes qu'au chapitre 2 :  $g_{Na}$ ,  $\tau_{act}$ ,  $\tau_{inact}$ ,  $E_{Na}$ ,  $V_{offset\_act}$ ,  $V_{offset\_inact}$ ,  $V_{pente\_act}$  et  $V_{pente\_inact}$ .

#### 3.1.1 Méthode

Les enregistrements de voltage-clamp ont été fournis par Gytis Baranauskas<sup>25</sup>. Ils ont été traités et publiés dans [BM06], en 2006.

Deux réseaux de six séries de courbes ont été enregistrés : l'un pour la caractérisation du canal sodique en réponse aux stimulations servant à déterminer la phase d'inactivation (fig 3.1, gauche), l'autre en réponse aux stimulations pour la phase d'activation (fig 3.1, droite). Les deux réseaux de mesures ont été effectués sur deux neurones différents et les impédances des électrodes nous étaient inconnues.

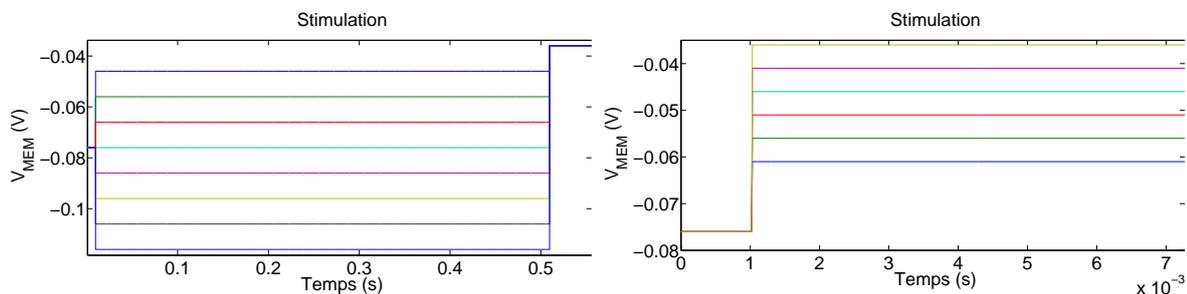


FIG. 3.1 – Enregistrement des courants en réponse au réseau de stimulations “inactivation”, à gauche, et “activation”, à droite.

<sup>25</sup>Instituto Italiano di Tecnologia, Italie

Malgré tout, afin d'estimer les huit paramètres en même temps comme s'il s'agissait d'un même neurone, nous avons mis à la même échelle les courants issus des deux ensembles d'enregistrements en multipliant par un facteur 85 les données relatives à la partie "activation". Ce facteur "pourrait" correspondre à l'impédance inconnue des électrodes ainsi qu'à une résistance membranaire différente d'un neurone à l'autre. Cependant ces hypothèses n'ont pas été testées pour notre étude. Nous avons ensuite estimé les paramètres simultanément comme présenté au chapitre précédent, en utilisant la méthode du DE avec, ici, un nombre d'individu  $NP = 300$  et  $N_{iter} = 250$ . 3900 points ont été utilisés sur chaque courbe pour réaliser l'optimisation. Tout comme au chapitre 2, les simulations ont été effectuées à l'aide de la version 7.7.0 du logiciel *Matlab* sur un PC *Dell* de processeur Intel Core duo<sup>26</sup> à 2.66GHz.

### 3.1.2 Résultats

Les résultats publiés dans [BM06] ont été obtenus avec la méthode d'estimation "classique" associée au voltage-clamp et une estimation différente pour les deux réseaux de courbes. Les courbes publiées dans [BM06] apparaissent figure 3.2, les paramètres n'ont en revanche pas été fournis. La comparaison des résultats entre la méthode "classique" et la méthode que nous avons développée est donc essentiellement visuelle.

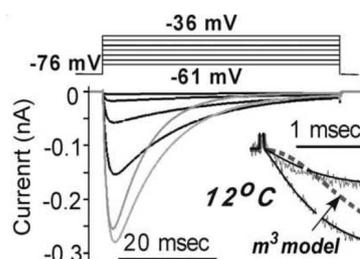


FIG. 3.2 – Courbes estimées (en noire) et enregistrées (en gris) pour la série "inactivation" (d'après [BM06]).

Notre méthode nous permet d'estimer simultanément tous les paramètres du modèle à constantes de temps fixes en prenant en compte directement les deux réseaux de courbes remis à la même échelle ; les valeurs sont regroupées dans le tableau 3.1. Les courbes, quant à elles, apparaissent à la figure 3.3. Environ une heure et demi a été nécessaire pour effectuer les 250 itérations. Cette durée d'estimation pourrait être considérablement réduite avec un code optimisé et/ou programmé en langage C.

Nous constatons que le modèle rend compte qualitativement des observations, mais qu'à la différence des résultats du chapitre 2, les courbes estimées et de référence ne sont pas parfaitement superposées. Il s'agit en effet, ici, de modéliser un comportement biologique, le modèle devant s'approcher le plus possible du comportement observé dans le vivant. Donc, dans l'absolu, ce modèle n'est pas "parfait", si tant est qu'il existe des modèles parfaits..., mais en regard des courbes obtenues à la figure 3.2 avec une méthode classique d'ajustement quasi-manuel, les résultats sont comparables, voire meilleurs (visuellement, l'erreur apparaît environ 10% plus

<sup>26</sup>Le code n'a pas été optimisé pour améliorer la vitesse des simulations sur une machine de type Core duo.

Paramètres	Valeurs estimées
$g_{ion} (\mu S)$	$1.7e^{-1}$
$\tau_m (ms)$	$2.9e^{-4}$
$E_{equi} (mV)$	$1.0e^{-4}$
$V_{offset\_m} (mV)$	$-5.6e^{-2}$
$V_{pente\_m} (mV)$	$5.6e^{-3}$
$\tau_h (ms)$	$1.8e^{-2}$
$V_{offset\_h} (mV)$	$-1.0e^{-1}$
$V_{pente\_h} (mV)$	$1.9e^{-2}$

TAB. 3.1 – Paramètres estimés par la méthode du DE à partir des données biologiques.

faible) avec la méthode d’ajustement automatisée à l’aide du DE. Cette application prouve donc la validité de notre méthode sur des données issues du vivant.

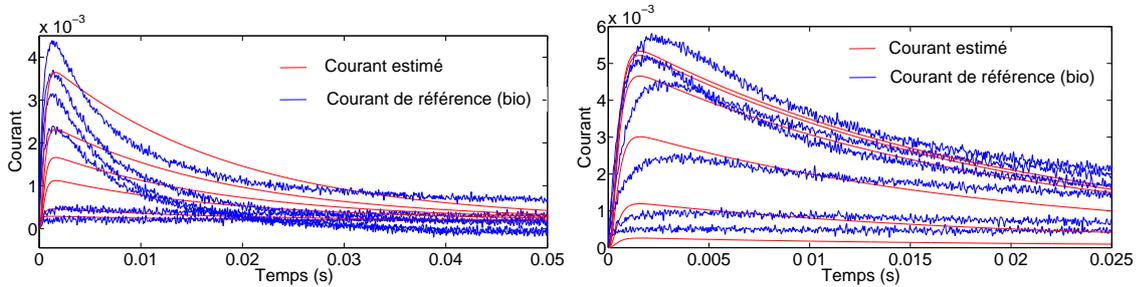


FIG. 3.3 – Courbes estimées et enregistrées pour les deux réseaux de courbes : à gauche, “inactivation” et à droite, “activation”.

Finalement, l’observation de ces résultats permet également de prouver la validité du modèle à constantes fixes ; du moins ce modèle ne rend-il pas moins compte des observations expérimentales que le modèle classique dans le cas de cette application.

## 3.2 Application au réglage de circuits neuromimétiques

Bien que conçus suivant les équations du modèle d’Hodgkin-Huxley à constantes de temps fixes, les circuits neuromimétiques de l’équipe ne répondent pas tous de la même façon à un même jeu de paramètres  $g_K$ ,  $g_{Na}$ ,  $E_K$ ,  $E_{Na}$ , etc. Ceci est dû à la dispersion des caractéristiques des composants liée au processus de fabrication. Il est donc nécessaire d’ajuster tous les paramètres pour chaque puce en fonction du comportement biologique à reproduire ou du modèle théorique à “imiter”. De plus, comme nous l’avons déjà expliqué au chapitre précédent, le modèle d’Hodgkin-Huxley est fortement non linéaire et le nombre important de ses paramètres rend impossible un réglage manuel des neurones électroniques, d’autant plus s’il s’agit de régler une centaine de neurones afin de les relier en réseau.

### 3.2.1 Méthode

Le fonctionnement des puces permet de réaliser des enregistrements de type voltage-clamp, nous avons donc appliqué notre méthode d’estimation des paramètres développée au chapitre

2 pour automatiser le réglage des circuits<sup>27</sup>. Nous décrivons ci-dessous brièvement le fonctionnement de la puce *Pamina* à laquelle nous avons appliqué pour la première fois la technique de réglage, ainsi que le principe de ce réglage.

### Le système électronique

La puce *Pamina* décrite ci-dessous a été réalisée en un seul exemplaire et regroupe plusieurs neurones. Elle a été développée par Sylvain Saïghi durant ces travaux de thèse (cf [Sai04]).

*Pamina* a été conçue en « full-custom<sup>28</sup> » à l'aide d'une technologie BiCMOS SiGe  $0.35\ \mu\text{m}$  du fondeur Austriamicrosystems (AMS), en utilisant la suite logicielle de CAO Cadence. Les modules chargés du calcul des courants ioniques ont été implémentés en mode courant, ce qui signifie que les variables des équations dérivées des modèles biophysiques sont physiquement représentées par des courants. *Pamina* comprend approximativement 19000 transistors MOS, 2000 transistors bipolaires et 1200 éléments passifs. La surface de la puce est de  $4170 \times 3480\ \mu\text{m}^2$ . Les générateurs de courant pour les canaux ioniques et les synapses ont été conçus en full-custom, ainsi que les cellules de mémoire analogique. Les cellules digitales utilisent le standard AMS ; elles ont été implémentées en utilisant un outil de placement et routage. Au final 71% des 22200 éléments sont placés et routés manuellement.

La figure 3.4 donne la description de la partie analogique du circuit intégré neuromimétique. Il regroupe les générateurs de courants ioniques suivant le modèle d'Hodgkin-Huxley décrit aux chapitres précédents. Afin de pouvoir modéliser un large type de modèles de neurones différents, nous avons choisi de représenter cinq types de canaux ioniques : la fuite, le sodium, le potassium, le calcium et le potassium dépendant calcium (dont nous pouvons choisir les conductances). Nous y avons ajouté huit synapses d'entrée et un générateur de courant de stimulation. La partie simulation analogique peut être configurée par l'intermédiaire de deux bus internes. Le premier –analogique– définit les valeurs des paramètres des équations 1.25 et 1.37 du chapitre 1. Ces paramètres ajustables un à un sont ensuite stockés sur les mémoires analogiques de la puce. Le second bus –numérique– spécifie quels seront les générateurs de courants ioniques activés lors de la simulation. La tension de membrane (via la sortie “ $V_{mem}$  pour visualisation”) ainsi que les différents courants ioniques redirigés vers la sortie peuvent être visualisés en temps réel individuellement. Ceci permet donc d'effectuer des mesures de type voltage-clamp et d'appliquer notre méthode d'optimisation.

Afin d'effectuer correctement les conversions des paramètres – des valeurs biologiques, vers des valeurs électroniques interprétables par le circuit –, la capacité de membrane de *Pamina* a été mesurée ; elle est égale à  $2,11\ \text{nF}$ .

Les plages de valeurs que peuvent prendre les paramètres sur la puce sont regroupées dans le tableau 3.2. Certains intervalles de valeurs étant très importants, deux sous-circuits multiplieurs ont dû être utilisés pour simuler les conductances des canaux de fuite et modulant. Ce qui signifie que nous avons dû estimer deux paramètres  $g_{fuite1}$ ,  $g_{fuite2}$ , au lieu d'un, pour la conductance de fuite, de même que pour la conductance modulante pour laquelle nous avons dû chercher  $g_{Mod1}$  et  $g_{Mod2}$ . En résumé, il a fallu rechercher les 16 paramètres stockés dans le vecteur  $X_{ion,ion=K,Na,fuite}$  pour reproduire le comportement d'un neurone FS (Fast Spiking<sup>29</sup>)

<sup>27</sup>La technique de réglage est ici appliquée à des circuits neuromimétiques suivant le modèle d'Hodgkin-Huxley ; néanmoins, une telle technique d'ajustement pourrait être appliquée à d'autres systèmes électroniques complexes nécessitant un réglage.

<sup>28</sup>Anglicisme courant spécifique de la conception électronique qui désigne une conception sur mesure avec positionnement manuel des éléments.

<sup>29</sup>Neurone dont le taux d'émission des potentiels d'action est élevé en comparaison avec d'autres neurones.

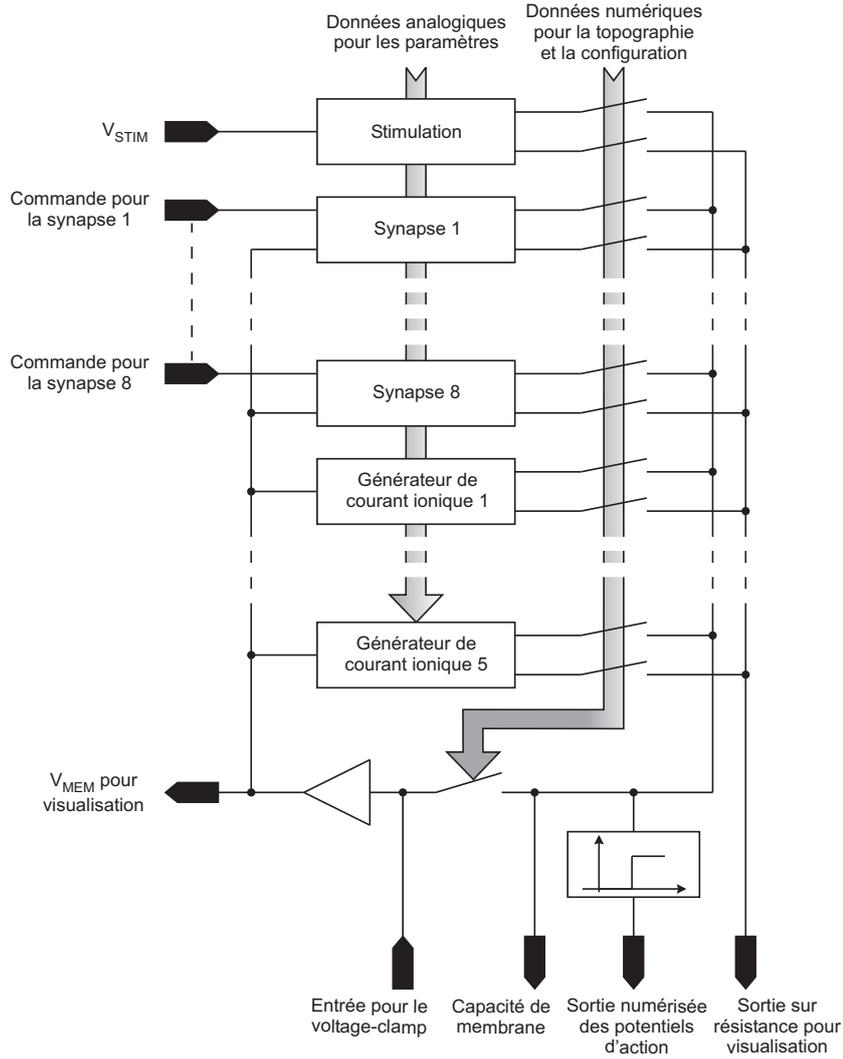


FIG. 3.4 – L’ASIC *Pamina*, d’après [Sai04].

avec le circuit neuromimétique :

$$X_K = [g_K \tau_m E_K V_{\text{offset}_m} V_{\text{pente}_m}], \quad X_{\text{fuite}} = [g_{\text{fuite}1} g_{\text{fuite}2} E_{\text{fuite}}],$$

$$X_{Na} = [g_{Na} \tau_m \tau_h E_{Na} V_{\text{offset}_m} V_{\text{offset}_h} V_{\text{pente}_m} V_{\text{pente}_h}].$$

Pour simuler un neurone RS (Regular Spiking<sup>30</sup>), 21 paramètres doivent être trouvés : les mêmes paramètres que ci-dessus ainsi que les éléments du vecteur

$$X_{Mod} = [g_{Mod1} g_{Mod2} \tau_m E_{Mod} V_{\text{offset}_m} V_{\text{pente}_m}]$$

Certains neurones du globus pallidus ou les interneurons corticaux inhibiteurs sont par exemple des neurones FS.

<sup>30</sup>Les neurones RS sont généralement continuellement actifs. Ils présentent une période dite d’adaptation avant de se stabiliser à une certaine fréquence d’oscillation dépendant de la stimulation ; d’où la présence d’un canal ionique modulant dans les équations qui les modélisent.

Parameters	<i>Na</i>	<i>K</i>	<i>Mod</i>	<i>fuite</i>
$g_{ion}(S)$	$[4,5e^{-6}; 2,27e^{-2}]$	$[4,5e^{-6}; 2,27e^{-2}]$	$[4,5e^{-7}; 2,27e^{-3}]$	$[2,27e^{-8}; 2,27e^{-3}]$
$\tau_m(ms)$	$[0,02; 2]$	$[0,2; 10]$	$[2; 50]$	
$E_{equi}(mV)$	$[100; 400]$	$[-600; -250]$	$[500; 650]$	$[100; 500]$
$V_{offset_m}(mV)$	$[-300; 0]$	$[-300; 0]$	$[-400; -100]$	
$V_{pente_m}(mV)$	$[10; 100]$	$[10; 100]$	$[10; 100]$	
$\tau_h(ms)$	$[-0,2; 2]$			
$V_{offset_h}(mV)$	$[-300; 0]$			
$V_{pente_h}(mV)$	$[10; 100]$			

TAB. 3.2 – Bornes des valeurs pouvant être prises par les paramètres sur la puce – valeurs électroniques pour une cellule de  $22.10^{-5}cm^2$ .

Le circuit intégré simule en temps réel la tension de membrane de chaque neurone. Pour effectuer les expériences, un système reliant *Pamina* à un ordinateur a été conçu par Sylvain Saïghi. Le développement du protocole de communication entre les différents appareils de mesures et l'ordinateur pour l'automatisation du réglage des circuits neuromimétiques a fait l'objet d'un stage de 2ème année par un élève de l'ENSEIRB, Wajdi Ben Salem, que nous avons co-encadré.

### Principe du réglage

Les données de référence utilisées pour l'optimisation sont des courants simulés en langage C à partir des équations du modèle à constantes de temps fixes et de paramètres fournis par Alain Destexhe<sup>31</sup> et disponibles dans les bases de données du site internet du logiciel *Neuron*. L'algorithme d'optimisation a été implémenté en C également. Pour la validation des résultats, le même modèle a été implémenté sous le logiciel *Neuron* pour simuler la tension de membrane et la réponse fréquentielle du neurone aux stimulations de voltage-clamp présentée au chapitre 1 (fig. 1.9, 1.10 et 1.11).

L'estimation des paramètres est effectuée canal ionique par canal ionique. Le courant de fuite  $I_{fuite}$  étant donné par une équation affine et ne mettant en jeu ni terme d'activation, ni terme d'inactivation, ses paramètres  $g_{fuite}$  et  $E_{fuite}$  sont estimés par une simple régression linéaire.

L'estimation des paramètres des autres courants ioniques, potassique, sodique et modulant, qui requièrent des méthodes d'optimisation plus complexes, est effectuée à l'aide de l'algorithme d'évolution différentielle que nous avons déjà présenté au chapitre 2. Nous proposons donc d'ajuster dynamiquement les paramètres de la puce afin de reproduire le modèle théorique dont les paramètres apparaissent au tableau 3.3 avec les résultats.

Comme dans les expériences de voltage-clamp effectuées in vitro et décrites au premier chapitre, nous appliquons des sauts de tension à la "membrane" du neurone électronique. Nous enregistrons, pour chaque saut de tension, noté  $stim$ , la tension délivrée par le générateur. Nous avons enregistré 9 sauts de tension pour chaque canal ionique, sauf le sodique pour lequel nous avons enregistré  $2 \times 9$  sauts de tension (correspondant à la série de stimulation "activation" et "l'inactivation"), et 500 points pour chaque courbe. Les données de référence calculées logicielle-ment à partir des sauts de tension enregistrés imposé au circuit et des paramètres de référence, sont notées  $I_{ion,ref}(t, stim)$ ; où  $t$  désigne le temps et  $ion$  représente successivement *K*, *Na* ou *Mod* pour le canal modulant.

<sup>31</sup>Unité de Neurosciences Intégratives et Computationnelles (UNIC), Gif-sur-Yvette, France

Le schéma de la figure 3.5 représente le principe de réglage d'un circuit neuromimétique avec la méthode d'estimation précédemment développée. Le principe de ce réglage automatique est le suivant :

1. Nous générons un jeu de paramètres suivant la méthode du DE et l'envoyons au circuit.
2. Le générateur de tension est commandé automatiquement par l'ordinateur pour l'envoi des sauts de tension.
3. Finalement, les réponses en courant du circuit,  $I_{\text{ion,VLSI}}(t, \text{stim})$ , sont enregistrées pour chaque saut de tension appliqué à la puce et comparées avec les réponses de référence précédemment calculées grâce à la fonction de coût  $F_{\text{coût}}$  que nous cherchons à minimiser :

$$F_{\text{coût}} = \sum_t \sum_{\text{stim}} (10^6 \times (I_{\text{ion,VLSI}}(t, \text{stim}) - I_{\text{ion,ref}}(t, \text{stim}))^2) \quad (3.1)$$

Remarquons que le facteur  $10^6$  est de nouveau présent pour éviter les limitations numériques.

4. Retour à l'étape 1.

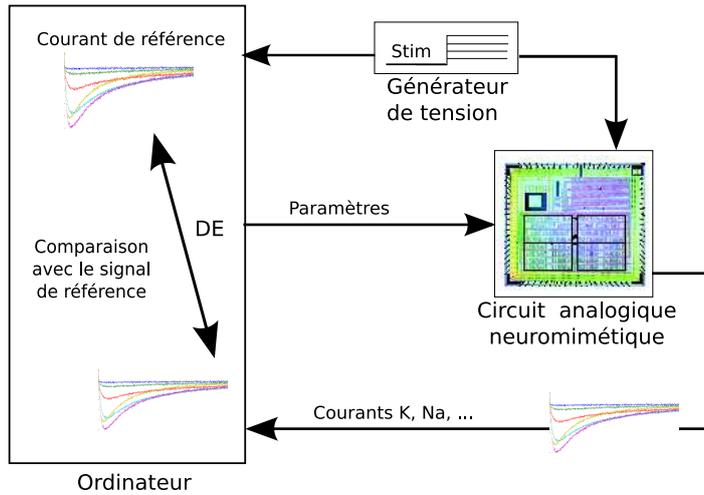


FIG. 3.5 – Principe du réglage automatisé des circuits neuromimétiques par ajustement des paramètres des courants ioniques.

Pour l'algorithme du DE, les paramètres suivants ont été choisis :  $NP = 50$  pour les courants potassique et modulants et  $NP = 60$  pour le courant sodique. Le critère d'arrêt a été défini expérimentalement comme suit ; l'estimation est considérée comme achevée si :

- Pour le potassium :  $N_{\text{iter}} = 150$  ou  $F_{\text{coût}} \leq 300$ .
- Pour le sodium :  $N_{\text{iter}} = 150$  ou  $F_{\text{coût}} \leq 900$ .
- Pour le canal modulant :  $N_{\text{iter}} = 150$  ou  $F_{\text{coût}} \leq 1.8$ .

Notons que le critère d'arrêt sur  $F_{\text{coût}}$  est différent pour chaque canal ionique car les courants n'ont pas tous le même ordre de grandeur.

Paramètres	Sodium		Potassium		Modulant		Fuite	
	Théor.	Estimés	Théor.	Estimés	Théor.	Estimés	Théor.	Estimés
$g_{ion} (\mu S)$	92,84	87,53	21,10	25,23	0,560	0,4075	0,330	0,310
$\tau_m (ms)$	0,07	0,029	1	0,4	61,25			
$E_{equi} (mV)$	250	236,3	-500	-601	559	232	400	389
$V_{offset\_m} (mV)$	172,1	202,1	145,4	191,1	206,4	606,2		
$V_{pente\_m} (mV)$	-32,35	-28,41	-39,27	-40,86	50,3	97,68		
$\tau_h (ms)$	0,362	0,144						
$V_{offset\_h} (mV)$	-195,35	- 213,06						
$V_{pente\_h} (mV)$	19,66	17,22						

TAB. 3.3 – Paramètres estimés et théoriques des canaux potassique, sodique et modulant – valeurs électroniques pour une cellule de  $22.10^{-5}cm^2$ .

### 3.2.2 Résultats – Discussion

Chaque estimation s’est achevée en atteignant le critère fixé par la valeur de la fonction de coût. Nous n’avons donc pas été limités par le nombre d’itérations. L’optimisation a duré environ 1h40 pour le potassium, 1h pour le modulant et 7h20 pour le sodium. Remarquons sur ce point que l’optimisation en elle-même est très rapide, mais qu’en revanche, la communication entre les différents appareils (ordinateur, puce, générateur) est beaucoup plus lente et ralentit le processus d’estimation.

Les caractéristiques de référence des canaux sodique et potassique étaient les mêmes pour le neurone FS et le neurone RS. Le tableau 3.3 regroupe les valeurs estimées après réglage. Une comparaison des courbes de courant de référence et obtenues avec les paramètres estimés apparaît à la figure 3.6.

On constate tout d’abord que l’écart entre les paramètres théoriques et estimés est très important sur le canal modulant. Ceci est en effet dû à l’ordre de grandeur du bruit, relativement important en comparaison de la faiblesse du courant modulant ; cette observation est corroborée par l’analyse de la figure 3.6. Cependant, en dépit de l’amplitude élevée du bruit, l’algorithme a su trouver une valeur moyenne de courant s’approchant des courbes de référence.

Lorsque l’on considère en revanche les canaux sodique et potassique, l’écart entre les paramètres est moins important : il varie entre 2 et 20% contre 20 à 95% pour le canal modulant (valeur la plus élevée pour  $V_{pente}$ ). De manière générale, les constantes de temps sont les paramètres qui nécessitent le plus un ajustement car l’écart entre la valeur théorique et la valeur à injecter à la puce est de l’ordre de 50% pour tous les canaux. Concernant les courbes estimées et enregistrées après réglage, celles-ci sont très proches.

Afin de nous assurer que le comportement du neurone, après un réglage canal par canal, était réellement proche de celui du modèle en ce qui concernait l’allure de la tension de membrane et sa réponse fréquentielle à des stimulations en courant, nous avons ensuite tracé la courbe de fréquence vs stimulation pour les neurones FS et RS, ainsi que la tension de membrane à une fréquence de 40 Hz environ pour le neurone FS<sup>32</sup> (figure 3.8).

Pour tracer la réponse fréquentielle de la figure 3.7, nous avons converti le courant de stimulation appliquée à la puce en valeurs biologiques en utilisant le ratio entre la capacité de

<sup>32</sup>Nous n’avons pas pu enregistrer la tension de membrane pour le neurone RS en raison de l’apparition soudaine et durable d’un problème de dysfonctionnement de la puce.

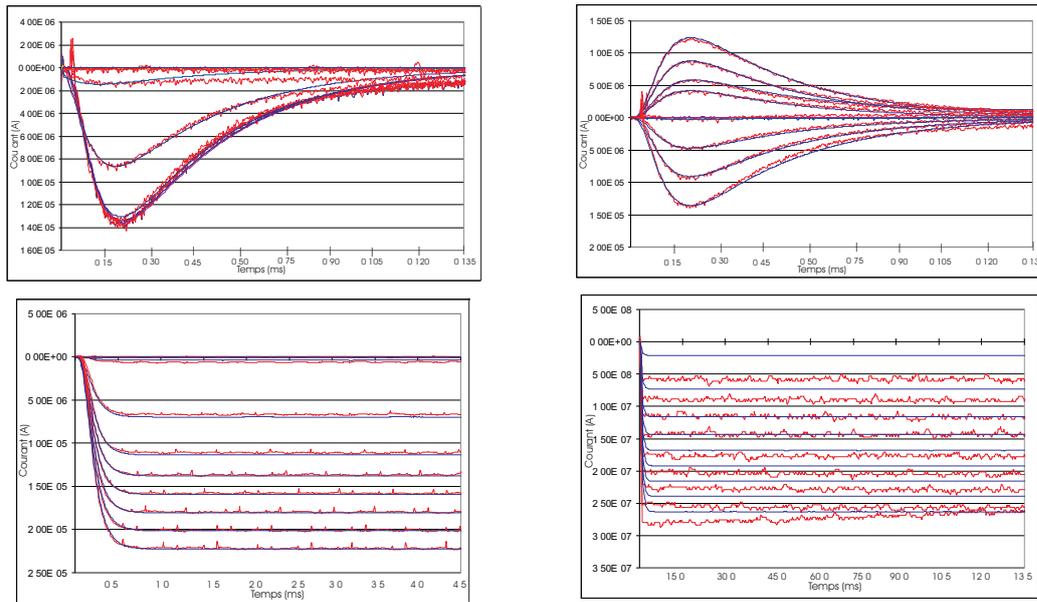


FIG. 3.6 – Réponses en courant des canaux sodique, potassique et modulant : courants théoriques et enregistrés après réglage. En rouge : courant VLSI (enregistré sur le circuit). En bleu : courant de référence.

membrane de la puce et celle du modèle. En fait, les courbes  $f(I)$  VLSI du neurone électronique ont été translatées de  $1.77nA$  et  $1.93nA$  pour les neurones FS et RS respectivement. Ces translations sont dues aux courants de fuite des composants de l'étage de sortie tels qu'ils existent typiquement dans tous les circuits analogiques intégrés. Les courbes du modèle et du neurone électronique sont presque totalement superposées, ce qui indique que, grâce à notre méthode automatisée d'ajustement des paramètres, le circuit est capable de reproduire le comportement fréquentiel du modèle biologique.

Concernant l'allure de la tension de membrane, le circuit est en mesure d'en reproduire les valeurs minimale et maximale cependant, la forme du potentiel d'action est légèrement différente autour de son seuil de déclenchement. Ce phénomène avait déjà été observé dans [Sai04] et semble être dû aux multiplieurs analogiques<sup>33</sup> et non à l'optimisation ou au choix des paramètres.

Les principales caractéristiques du modèle sont donc reproduites avec la puce, en particulier la réponse fréquentielle du neurone qui s'avère très importante lorsque nous ajoutons des règles de plasticité comme la STDP (Spike Timing Dependent Plasticity) qui en dépendent fortement ??.

Cette technique automatisée de réglage de circuits intégrés neuromimétiques est donc maintenant utilisée dans l'équipe pour ajuster les paramètres des nouvelles puces en fonction des comportements désirés et des applications visées. Une fois chaque neurone "réglé", il nous sera possible de les connecter en réseau afin d'en étudier les comportements en fonction de sa structure et de la dynamique individuelle des cellules qui le constituent. Le chapitre 4 a pour objet la préparation de cette étude.

<sup>33</sup>En effet, lorsqu'il s'agit, en électronique de multiplier un courant par zéro, le courant "égal" à zéro est en réalité un courant très faible, mais non nul, d'où une erreur.

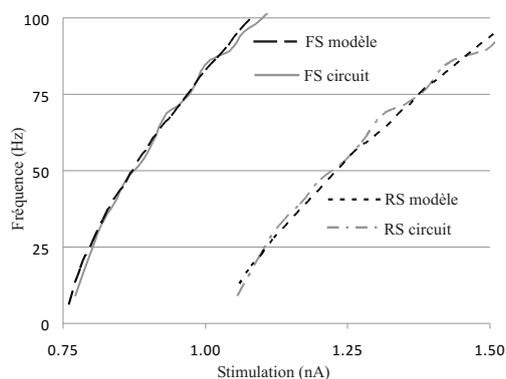
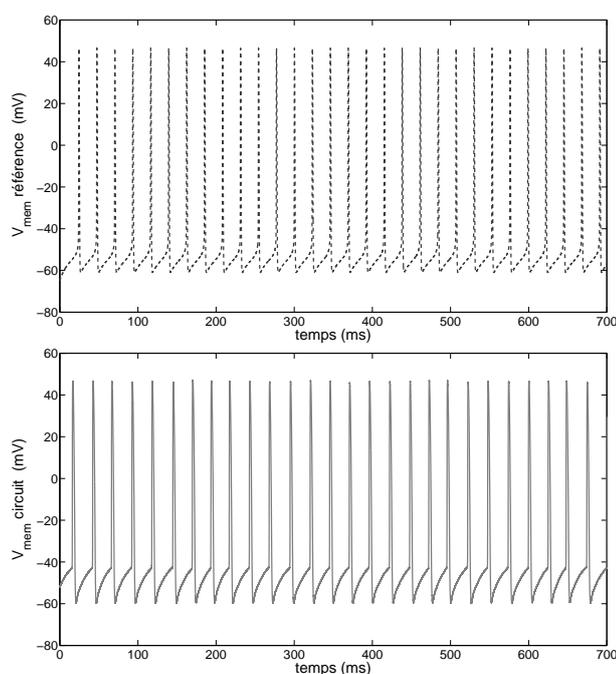
FIG. 3.7 – Comparaison des courbes  $f(I)$  pour les neurones FS et RS.

FIG. 3.8 – Tension de membrane à 45 Hz après réglage du neurone FS. Ligne pointillés : réponse du modèle. Ligne continue : réponse du neurone électronique.

Par ailleurs, ce réglage devrait nous permettre d'étudier la disparité d'un circuit à l'autre, d'évaluer la dispersion des caractéristiques des composants électroniques, en réalisant des statistiques sur la dispersion des différents paramètres ajustés et leur éloignement aux valeurs de référence.

### 3.3 Estimation des paramètres d'un modèle à partir de la tension de membrane d'un neurone

L'étude a été menée sur les circuits analogiques neuromimétiques de l'équipe car ils permettent une simulation en temps réel ou accélérée de l'activité d'un neurone. De cette façon, l'estimation des paramètres n'est limitée, en terme de rapidité, que par le temps de communication entre l'ordinateur, les appareils de mesure, le circuit neuromimétique et le temps d'exécution de l'algorithme d'optimisation. Cependant, le développement de la méthode d'estimation est resté inachevé car nous avons rencontré des problèmes de dysfonctionnement des circuits et la nouvelle génération de puces n'était pas encore disponible. Par conséquent, nous ne présentons ici que la méthode et des résultats partiels pour un modèle de neurone de type Fast Spiking, présentant une activité quasi-périodique (en présence d'une stimulation constante) et sans phase d'adaptation. L'estimation des paramètres pour un modèle de neurone Regular Spiking, estimation qui était en cours d'élaboration lorsque nous avons rencontré les problèmes techniques, n'a donc pas pu être réalisée.

#### 3.3.1 Méthode

Comme pour tout problème d'optimisation, il a fallu choisir un modèle, des données de référence, une fonction de coût et un algorithme d'optimisation que nous détaillons ci-dessous.

##### Le modèle

Comme dans les chapitres précédents, le modèle choisi est le modèle d'Hodgkin-Huxley en raison de son utilisation dans l'équipe. Néanmoins, cette méthode d'optimisation n'utilisant pas de "portion" d'équations spécifiques au modèle – comme c'était le cas avec les courants ioniques – nous pourrions l'utiliser également pour estimer les paramètres de tout autre modèle de neurone dans lequel est exprimée la valeur de la tension de membrane comme dans [RGPB10].

A défaut d'enregistrements biologiques et afin de pouvoir évaluer par la suite le mismatch des composants électroniques sur le circuit neuromimétique que nous souhaitons régler, nous avons utilisé comme données de références des tensions de membrane simulées grâce au logiciel *Neuron* avec lequel nous avons implémenté les équations du modèle d'Hodgkin-Huxley implantées dans les circuits intégrés. Comme il existait une incertitude sur la valeur de la stimulation imposée à la "membrane" du neurone artificiel, nous avons également estimé la valeur du courant de stimulation qui faisait partie des paramètres recherchés au même titre que  $g_K$ ,  $g_{Na}$ ,  $g_{L1}$ ,  $g_{L2}$ ,  $E_K$ ,  $E_{Na}$ ,  $E_L$ ,  $\tau_n$ ,  $\tau_h$ ,  $\tau_m$ ,  $\tau_h$ ,  $V_{pente\_n}$ ,  $V_{pente\_m}$ ,  $V_{pente\_h}$ ,  $V_{offset\_n}$ ,  $V_{offset\_m}$ ,  $V_{offset\_h}$ , pour un neurone de type FS (voir le chapitre 1, section 3, et le chapitre 3, section 2, pour un rappel sur la signification des paramètres).

##### Choix d'une fonction de coût

Contrairement à l'estimation réalisée à partir d'enregistrements de courants ioniques pour laquelle seule la réponse du courant dans le temps nous importait, cette optimisation nécessite de s'intéresser au comportement dynamique de la cellule. En effet, le signal à reproduire est plus complexe et nécessite la prise en compte de plusieurs critères que nous combinons dans une unique fonction (mono-objectif).

Pour la construction de la fonction de coût, nous nous sommes donc intéressés tout d'abord à l'activité du neurone dans le plan de phase  $\Phi = \{(dV_{MEM}/dt, V_{MEM})\}$  et nous avons calculé le  $dV_{MEM}/dt$  moyen sur plusieurs périodes pour une simulation d'une seconde (lorsque la fréquence

des potentiels d'action du modèle de référence était supérieure à 2 Hz). L'échantillonnage de l'enregistrement était de 50000 points par seconde. Afin de pouvoir manipuler des valeurs entières de la tension de membrane, nous avons multiplié toutes les tensions électroniques mesurées en  $mV$  par 100 et travaillé en virgule fixe ; d'où une précision au centième de millivolt, largement suffisante en comparaison du bruit électronique, de l'ordre de quelques dixièmes de millivolts. Une partie du coût se calculait donc ainsi :

$$(dV_{\text{MEM, ref}} - dV_{\text{MEM, est}})^2$$

Où "est" représente le signal enregistré depuis la puce avec les paramètres estimés et "ref" désigne le signal de référence.

Ce critère permet de reproduire la forme du potentiel d'action, mais est insuffisant pour prendre en compte la fréquence du signal. Nous avons alors ajouté, pour le calcul du coût, la norme  $\mathcal{L}^1$ ,  $|N_{\text{spike, ref}} - N_{\text{spike, est}}|$ , de la différence entre le nombre de spikes du signal de référence et du signal estimé à l'aide de la puce. Finalement, après quelques tests d'efficacité, nous avons constaté que l'algorithme était plus performant (en terme de nombre d'itérations nécessaires et de qualité de la solution) en ajoutant un troisième critère : l'erreur quadratique,  $(A_{\text{ref}} - A_{\text{est}})^2$ , entre les amplitudes des potentiels d'action de référence et estimés. La fonction de coût,  $\varphi$ , que nous avons implémentée pour un neurone FS a donc la forme suivante :

$$\varphi = a.(dV_{\text{MEM, ref}} - dV_{\text{MEM, est}})^2 + b.|N_{\text{spike, ref}} - N_{\text{spike, est}}| + c.(A_{\text{ref}} - A_{\text{est}})^2,$$

où  $a$ ,  $b$  et  $c$  sont des facteurs de pondération ajustés empiriquement.

Notons enfin qu'il est nécessaire de prendre en compte des critères supplémentaires lorsqu'il s'agit d'estimer les paramètres de neurones exhibant des comportements plus complexes tels que l'adaptation du RS, la période et la durée des bursts pour un neurone de type LTS, etc. Dans ces derniers cas, nous avons simplement ajouté la norme  $\mathcal{L}^2$  de la différence entre la caractéristique du signal de référence et la caractéristique obtenue avec le circuit. Par exemple, pour un neurone RS, les caractéristiques supplémentaires sont les trois premiers intervalles interspikes après stimulation,  $I_1$  entre les deux premiers potentiels d'action,  $I_2$  entre le deuxième et le troisième,  $I_3$  entre le troisième et le quatrième, et les trois derniers intervalles,  $I_{\text{fin}1}$ ,  $I_{\text{fin}2}$ ,  $I_{\text{fin}3}$  (caractéristiques similaires à celles utilisées dans [PTRM<sup>+</sup>08]). La nouvelle fonction de coût  $F_{\text{coût\_total}}$  devient donc :

$$F_{\text{coût\_total}} = \varphi + \sum_{i=1, \dots, 3} (I_{i, \text{ref}} - I_{i, \text{est}})^2 + \sum_i (I_{\text{fin} i, \text{est}} - I_{\text{fin} i, \text{ref}})^2$$

où ref désigne la valeur pour la donnée de référence et est pour la donnée estimée.

Comme cette étude n'a pas été menée à son terme, nous n'avons implémenté la méthode que pour estimer les paramètres du modèle en réponse à une valeur de stimulation, mais il serait plus pertinent d'effectuer l'optimisation pour plusieurs valeurs différentes du courant de stimulation.

### L'algorithme d'optimisation

Dans les chapitres précédents, l'algorithme d'évolution différentielle a prouvé son efficacité, nous avons donc choisi de le tester sur ce nouveau problème.

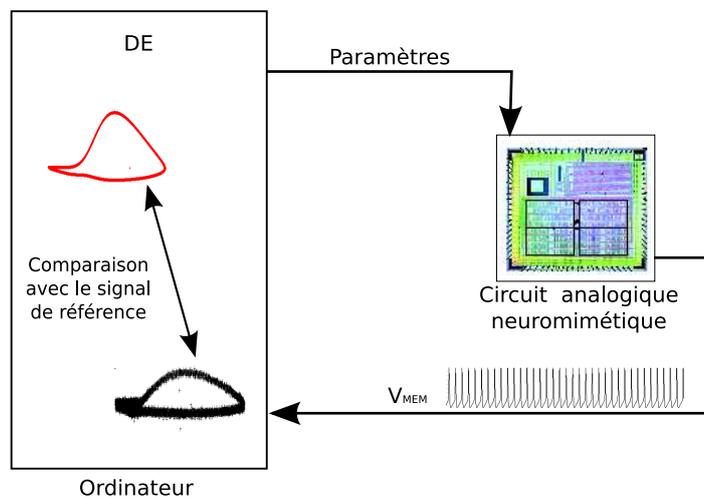


FIG. 3.9 – Schéma représentant le principe de réglage automatique de *Pamina* à partir de l'enregistrement de la tension de membrane.

### Implémentation de la méthode

Le principe est le même que dans le chapitre 3 et résumé dans le schéma qui suit (voir fig. 3.9) :

1. Un jeu de paramètres est généré sur un ordinateur suivant l'algorithme d'évolution différentielle ;
2. Ce jeu de paramètres est ensuite envoyé au circuit neuromimétique ;
3. La réponse du neurone artificiel est enregistrée<sup>34</sup> et comparée à la réponse du neurone de référence au moyen de la fonction de coût.
4. le jeu de paramètre est ensuite conservé ou modifié selon l'algorithme du DE, d'après la valeur de la fonction de coût.

Et ainsi de suite jusqu'à atteindre le critère d'arrêt correspondant à un nombre maximal d'itérations.

L'algorithme d'optimisation a été implémenté en C/C++ (gcc), sur un PC *Dell* de processeur Intel Core duo à 2.66 GHz, comportant 3 Go de RAM, avec la version 4.0 de la distribution Debian.

Le circuit intégré utilisé pour les simulations matérielles est le circuit *Pamina* développé par Sylvain Saïghi durant ses travaux de thèse [Sai04]. Il peut ici être vu comme un co-processeur dédié au calcul de la tension de membrane.

### 3.3.2 Quelques résultats

Nous présentons dans cette section uniquement les courbes de référence et les courbes obtenues sur le circuit après réglage : la figure 3.10 représente les courbes dans le plan de

<sup>34</sup>Il est possible de visualiser simultanément l'évolution de la tension de membrane sur un oscilloscope

phase  $\Phi = \{(dV_{\text{MEM}}/dt, V_{\text{MEM}})\}$  et la figure 3.11 la tension de membrane en fonction du temps.

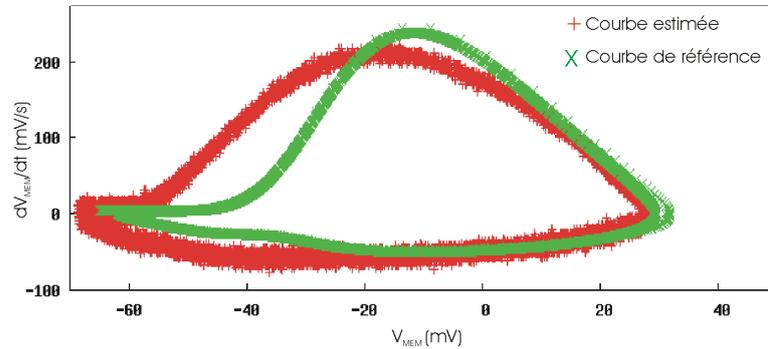


FIG. 3.10 – Comparaison des diagrammes de phase.

On constate, sur les deux courbes, que le seuil de déclenchement du potentiel d'action et la valeur minimale de la tension de membrane sont inférieurs ( $-55 \text{ mV}$  et  $-65 \text{ mV}$ ), sur les enregistrements issus de la puce, à ceux provenant du modèle ( $-42 \text{ mV}$  et  $-62 \text{ mV}$ ). La valeur maximale de la tension de membrane est également légèrement plus élevée avec le modèle qu'avec le circuit. Ces écarts avaient déjà été observés lorsque les paramètres du circuit étaient ajustés grâce aux enregistrements des courants ioniques [BSG<sup>+</sup>09]. Il est donc possible que les différences observées proviennent des caractéristiques du circuit et que la solution trouvée soit optimale.

Cependant, la fréquence d'oscillation est parfaitement reproduite (avec ici,  $92 \text{ Hz}$ ). Des résultats similaires ont été observés avec des fréquences plus faibles. Ces premiers résultats sont donc très encourageants et promettent le développement d'un outil permettant de reproduire l'activité de divers types de neurones en appliquant la méthode à des cellules ayant une dynamique plus complexe. Finalement, afin de pousser plus loin l'analyse, il serait intéressant de s'assurer, à l'inverse des résultats présentés au chapitre 3, que le fait de reproduire le comportement dynamique et fréquentiel de la tension de membrane soit, dans notre cas, une condition suffisante pour reproduire les comportements des canaux ioniques du modèle. Cette vérification serait possible en effectuant, après réglage, des mesures en mode voltage-clamp sur le circuit neuromimétique.

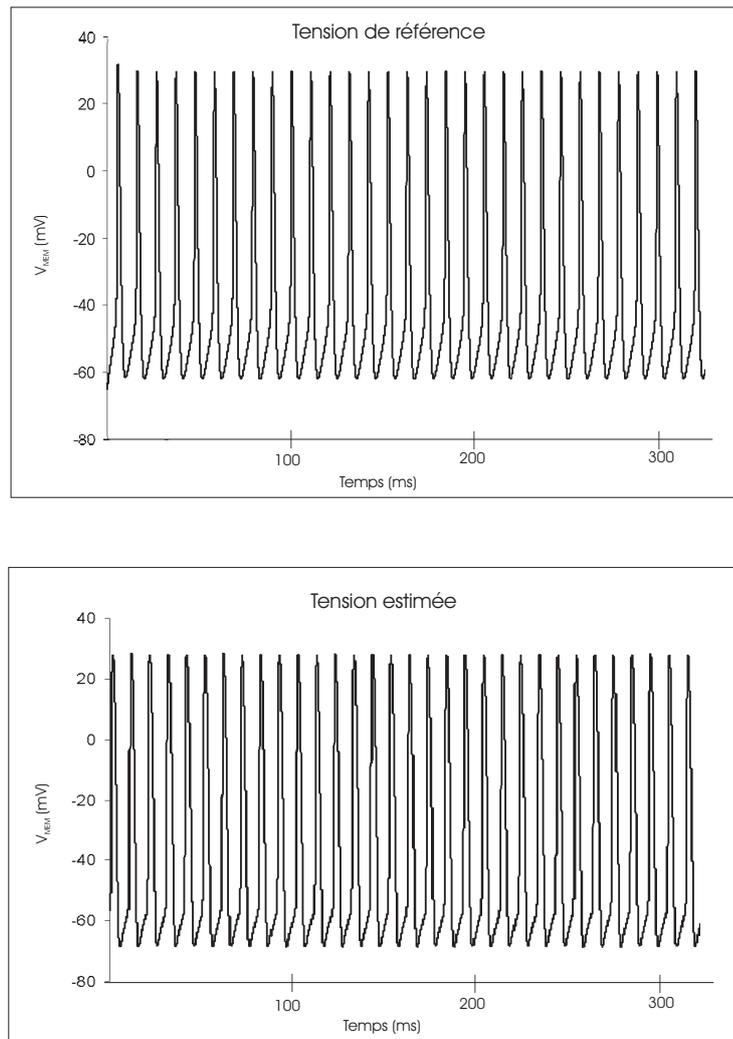


FIG. 3.11 – Comparaison des courbes de tension de membrane en fonction du temps.

## 3.4 Résumé

Ce chapitre regroupe deux types d'applications de la méthode d'estimation de paramètres de modèles neuronaux présentée au chapitre précédent. La première concerne l'extraction d'un modèle de neurone à partir d'enregistrements de courants ioniques en voltage-clamp (données fournies par Gytis Baranauskas, Instituto Italiano di Tecnologia, Italie). La méthode n'est appliquée qu'au courant sodique car ce courant est le plus complexe à caractériser et l'utilité de la méthode apparaît ainsi clairement. Les courbes obtenues avec notre méthode sont comparées à celles obtenues avec la méthode classique associée au voltage-clamp et mettent en évidence l'efficacité de notre technique d'estimation conjointe de tous les paramètres du canal sodique. La seconde application a son entière utilité au sein de notre équipe de recherche : il s'agit d'une méthode automatisée de réglage de circuits neuromimétiques à partir d'enregistrements de type voltage-clamp effectués sur ces circuits. Les puces sont réglées canal ionique par canal ionique et exhibent, après réglage, un comportement fréquentiel semblable à celui du modèle. La tension de membrane du neurone électronique enregistrée après réglage est également très proche de celle du modèle. La méthode automatique de réglage est donc efficace et est aujourd'hui employée pour ajuster les paramètres des nouvelles puces en fonction du comportement souhaité en vue de la constitution de réseaux de neurones électroniques. Après nous être intéressés à l'ajustement des paramètres d'un modèle de neurone à partir des courants ioniques, nous cherchons à estimer les paramètres directement à partir d'enregistrements de la tension de membrane, signaux plus complexes, mais dont l'acquisition en neurophysiologie est plus aisée que celles des courants ioniques. Nous avons utilisé, pour développer la méthode, des circuits neuromimétiques développés dans l'équipe suivant le modèle d'Hodgkin-Huxley et dont nous souhaitions régler les paramètres. Ces circuits peuvent donc être vus comme des co-processeurs dédiés au calcul de la tension de membrane. Les données de référence étaient des tensions de membrane simulées à l'aide du logiciel *Neuron*. En raison de problèmes techniques sur les circuits neuromimétique cette étude n'a pu être menée à terme et seule la technique de réglage automatique pour un neurone de type FS a été développée. Elle est présentée en troisième section du chapitre.



## Chapitre 4

# Vers la configuration d'un réseau de neurones

### Sommaire

---

<b>4.1</b>	<b>Intoduction</b> . . . . .	<b>86</b>
<b>4.2</b>	<b>Modèle et méthodes</b> . . . . .	<b>88</b>
<b>4.3</b>	<b>Les différents réseaux de neurones envisagés</b> . . . . .	<b>91</b>
4.3.1	Réseau de neurones excitateurs et inhibiteurs de type I ou II exclusivement	91
4.3.2	Réseau mixte de neurones excitateurs de type I et d'inhibiteurs de type II	100
4.3.3	Réseau de neurones inhibiteurs . . . . .	101
<b>4.4</b>	<b>Conclusion</b> . . . . .	<b>104</b>
<b>4.5</b>	<b>Résumé</b> . . . . .	<b>106</b>

---

« Μηδέν Ἄγαν »  
« Rien de trop » (Platon, Philèbe, 40a)

Dans les précédents chapitres nous nous sommes attachés à décrire et appliquer une méthode permettant d'estimer les paramètres d'un modèle de neurone, le modèle d'Hodgkin-Huxley, à partir de l'enregistrement de courants ioniques. Cette technique, qui a donc été validée logiquement et sur des données expérimentales, peut également servir au réglage de circuits neuromimétiques. Nous avons ensuite proposé, afin de nous adapter aux besoins des neurophysiologistes, une méthode d'estimation des paramètres du même modèle à partir d'enregistrements de la tension de membrane, plus aisés à réaliser pour les biologistes. Ce dernier aspect nous a poussé à prendre en compte des aspects plus globaux de la dynamique du neurone, mais sans jamais passer à l'échelle du réseau de neurones. Or, l'un des buts de la modélisation neuronale est de pouvoir simuler le comportement de réseaux de neurones dont l'enregistrement de l'activité dans le vivant reste très complexe ; en ce sens l'utilisation de réseaux de circuits analogiques neuromimétiques peut constituer une alternative d'étude. La simulation d'un réseau, comme son nom l'indique, met donc en jeu plusieurs cellules reliées entre elles, chacune ayant ses caractéristiques propres. On peut alors se demander quelle sera l'influence de la dynamique individuelle des neurones sur le comportement global du réseau. Une étude théorique tentant de répondre à cette question est donc préalable à l'implémentation d'un réseau de neurones artificiels à l'aide de circuits neuromimétiques. Ce chapitre s'intéresse donc à cette vaste problématique, mais uniquement dans le cadre de la génération d'oscillations gamma. Cette partie du travail de thèse a été réalisée au Courant Institute of Mathematical Sciences / Center for Neural Science à l'Université de New York de mi-Janvier à mi-Avril 2010 sous la supervision de John Rinzel, dans le cadre du programme de stage proposé par l'Advanced Course in Computational Neuroscience.

## 4.1 Introduction à l'étude de l'influence des propriétés intrinsèques d'un neurone sur la dynamique globale du réseau – Oscillations gamma

Nous nous sommes jusqu'ici intéressés à l'ajustement des paramètres de modèles pour une cellule uniquement, or l'objectif de la modélisation neuronale est également de pouvoir mettre en réseaux plusieurs neurones. La question se pose alors de savoir quelle sera l'influence des propriétés intrinsèques du neurone sur le comportement du réseau, quelles seront les dynamiques à reproduire et quelle importance accorder aux motifs oscillatoires observés. Nous tenterons dans la suite de ce manuscrit de répondre à la première question en nous focalisant sur l'étude d'un motif oscillatoire spécifique : les oscillations gamma, oscillations entre 20 et 80 Hz pouvant être observées dans certains réseaux de neurones.

Bien que la question de la nature de l'information neuronale dans le cerveau soit encore totalement ouverte, certains types d'activités de réseaux neuronaux ont déjà été isolés et étudiés. Ces motifs d'activité semblent être une source d'information "compréhensible" par le cerveau. Par exemple, Gray et Singer montrent dans [GS89] que ce type d'oscillations peut jouer un rôle dans la transmission d'informations visuelles. La revue [?] publiée par Bartos et al. en 2007, décrit elle aussi l'intérêt des oscillations gamma dans différentes aires cérébrales comme celles liées à la formation de la mémoire. La dynamique neuronale à laquelle nous allons en particulier nous intéresser dans cette partie est donc la génération d'oscillations gamma dans un petit réseau de neurones.

Plusieurs études ont mis en évidence le fait que l'inhibition synaptique permet de synchroniser l'activité d'un réseau de neurones. Des travaux ont été menés concernant les réseaux d'interneurones (connexions gaba-ergiques) et l'apparition d'activité dans la plage de fréquences des

oscillations gamma, ie entre 20 Hz et 80 Hz : ces oscillations ont en particulier été observées dans l'hippocampe [BC95] et [CJWB03], dans le cortex entorhinal du rat [CB98], et reproduites à l'aide de modèles par Wang et Buzsáki dans [WB96] ou Kopell et al. dans [KE04]. Les réseaux mixtes, constitués à la fois de cellules excitatrices et inhibitrices, sont également capables d'engendrer des oscillations gamma (cf. [WSMS01] et [HS05] dans le cortex visuel avec modélisation à l'aide de neurones IF ou [ODRR09] dans le cortex auditif). Dans tous les cas sus-cités, les auteurs utilisaient systématiquement des modèles de neurones capables de reproduire des dynamiques de type I. Par ailleurs, des études ont été faites sur les dynamiques de type I et II de neurones biologiques isolés comme dans [THR04] et ont montré que les neurones excitateurs exhibaient généralement des comportements de type I et les neurones inhibiteurs des dynamiques de type II. L'une des principales caractéristiques des dynamiques de type I est la continuité de la réponse fréquentielle du neurone à une stimulation ; au contraire un neurone de type II ne peut pas osciller en deçà d'une certaine fréquence<sup>35</sup>. Les courbes des figures 4.3 et 4.6 illustrent cette observation.

L'idée principale de l'étude qui va suivre était d'analyser l'influence des propriétés intrinsèques d'un neurone sur le comportement général du réseau. Pour cela, nous avons modélisé numériquement sur logiciel plusieurs réseaux de 100 neurones en choisissant des paramètres différents pour les neurones excitateurs et les neurones inhibiteurs : i) chacune des deux populations étant de type I uniquement (respectivement de type II uniquement), ii) neurones excitateurs modélisés par des cellules de type I et inhibiteurs par des cellules de type II ; iii) 100 neurones de type II uniquement inhibiteurs. Dans le but d'implémenter par la suite ces réseaux sur des circuits neuromimétiques, naturellement bruités – comme nous l'avons déjà évoqué – nous avons également introduit du bruit, dans la stimulation, et de l'hétérogénéité dans chaque modèle de cellule (afin que tous les neurones soient légèrement différents les uns des autres).

Les oscillations gamma sont observables lorsque l'activité du réseau se synchronise (au moins partiellement) en réponse à une stimulation non nécessairement périodique. Wang et Buzsáki ont montré dans [WB96] que l'apparition d'une synchronisation dans le réseau n'était pas en relation avec la proportion de cellules connectées au sein de ce dernier, mais plutôt à un nombre moyen fixe de connexions par neurone. En effet, dans leur modèle, chaque neurone du réseau doit être en moyenne connecté à au moins 60 cellules, et ce que le réseau soit constitué de 100, 200, 500 ou 1000 neurones. Lorsque le nombre de synapses est inférieur à 60, il est alors impossible (ou très fortement dépendant de la connectivité) d'observer une synchronisation. Par ailleurs, le couplage inhibiteur doit être suffisamment important (en terme de conductance synaptique) pour pouvoir observer des oscillations du réseau ; ou, de manière équivalente, l'hétérogénéité des cellules ne doit pas être trop importante, en particulier pour les cellules excitatrices lorsqu'il s'agit de réseaux mixtes inhibiteurs/excitateurs. Dans le cadre de réseaux strictement inhibiteurs, White et al. [WCR<sup>+</sup>98] ont exploré la robustesse de la synchronisation lorsque le réseau est totalement connecté (tous les neurones sont connectés les uns aux autres) ; Bartos et al. ont montré quant à eux que dans les modèles de réseaux d'interneurones, il était nécessaire d'ajouter un court délai synaptique aux connexions inhibitrices pour pouvoir observer une synchronisation à condition que la synapses aient elle aussi des constantes de temps courtes (croissance  $< 1\text{ ms}$ , décroissance  $< 10\text{ ms}$ ). Nous avons donc pris en compte ces observations dans la modélisation de nos réseaux.

<sup>35</sup>La définition des deux types de dynamiques est en fait liée à l'existence d'une bifurcation d'Andronov-Hopf (Type II) ou d'un point selle, dit également point col, (Type I) dans un plan de phase défini par deux des variables du modèle. Nous ne rentrons pas ici dans les détails de la définition de ces deux types et ne retenons que le comportement de la réponse fréquentielle de chacun.

## 4.2 Modèle et méthodes

Cette section présente le modèle et les outils logiciels et d'analyse utilisés pour la modélisation de nos réseaux et l'étude de leurs comportements.

### Le modèle du réseau

Concernant la modélisation d'un neurone à l'échelle de la cellule, nous avons choisi un modèle nous permettant de reproduire plusieurs types de comportements neuronaux (FS, RS, LTS, etc) avec différentes dynamiques (type I, type II, intégrateur, résonateur...) que nous définissons plus loin. Nous avons de plus opté pour un modèle dont les comportements dynamiques avaient été bien étudiés et décrits en fonction du choix des paramètres<sup>36</sup> (voir figure 4.1). Les termes « intégrateur » et « résonateur » font ici référence au comportement du neurone sous le seuil de déclenchement du potentiel d'action : le premier intègre la stimulation lorsqu'elle augmente, jusqu'au déclenchement du potentiel d'action, alors que le second oscille sous le seuil de déclenchement.

Ce modèle, dit « adaptatif exponentiel Integrate and Fire », a été introduit par Romain Brette et Wulfram Gerstner dans [BG05]. Une étude de l'activité dynamique en fonction des plages de paramètres a été publiée dans [TB08]. Ce modèle est défini comme suit :

$$C \frac{dV}{dt} = -g_L(V - E_L) + g_L \Delta_T e^{\frac{V - V_T}{\Delta_T}} - w + I_{\text{app}} \quad (4.1)$$

$$\tau_w \frac{dw}{dt} = a(V - E_L) - w \quad (4.2)$$

$$\text{lorsque } V > 20 \text{ mV (ie spike) : } \begin{aligned} V &\rightarrow E_L \\ w &\rightarrow w + b \end{aligned} \quad (4.3)$$

où  $g_L$  désigne la conductance de fuite,  $E_L$  son le potentiel d'équilibre,  $\Delta_T$  représente la pente déterminant la largeur du seuil du potentiel d'action et  $V_T$  est le potentiel du seuil. Ce modèle est dit adaptatif en raison du terme  $w$  qui représente un courant d'adaptation nécessaire pour reproduire toute la variété d'activité des différents types de neurones. Le facteur  $a$  s'exprime en siemens. L'élément  $b$  est nécessaire pour l'adaptation après un potentiel d'action (spike-triggered adaptation); par exemple, une large valeur de  $b$  conduit à une forte adaptation<sup>37</sup>. Finalement,  $I_{\text{app}}$  désigne une éventuelle stimulation (en ampères).

Les valeurs des paramètres que nous avons fixées sont les suivantes :  $C = 281 \text{ pF}$ ,  $g_L = 30 \text{ nS}$ ,  $E_L = -70,6 \text{ mV}$ ,  $V_T = -50.4 \text{ mV}$ ,  $\delta_T = 2 \text{ mV}$ ,  $V_r = -70.6 \text{ mV}$ . Les valeurs des autres paramètres dépendent du type de neurone que nous utilisons pour chaque simulation.

Les synapses sont modélisées par un saut, puis une décroissance exponentielle de la conductance synaptique de la façon suivante :

$$dg_{\text{syn}} = -\frac{g_{\text{syn}}}{\tau_{\text{syn}}}$$

<sup>36</sup>Nous remarquerons qu'ici, les neurones sont différenciés suivant deux classes, A et B, et non les deux types habituels, I et II. En effet, la bifurcation mise en jeu dans ce modèle est une bifurcation de Bogdanov-Takens; en revanche, on retrouve bien des neurones de type II parmi les neurones de classe B et des neurones de type I parmi les neurones de classe A.

<sup>37</sup>Rem : Le seuil de déclenchement du potentiel d'action pour un courant de stimulation constant est donné par  $V_{\text{thresh}} = V_T + \Delta_T \log(1 + a/g_L)$  pour un neurone de classe A et par  $V_{\text{thresh}} = V_T + \Delta_T \log(1 + \tau_m/\tau_w)$  pour un neurone de classe B

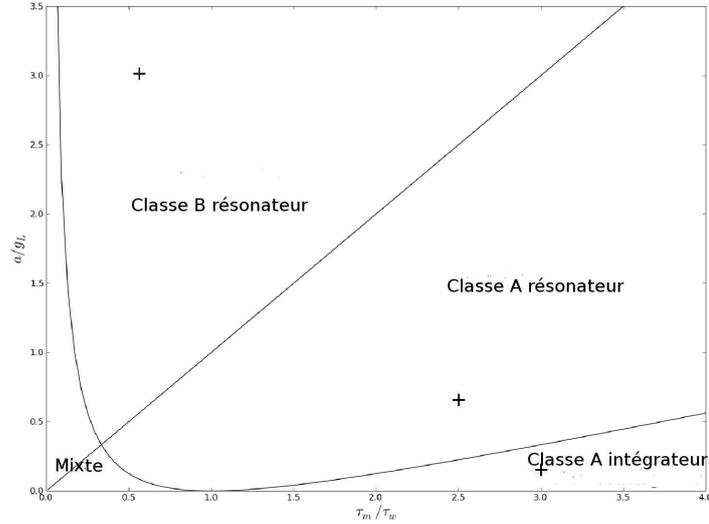


FIG. 4.1 – Diagramme montrant les séparatrices entre les classes A et B, et les neurones de type intégrateur et résonateur. Les croix correspondent aux valeurs de paramètres que nous avons choisies pour les simulations. D’après [TB08].

où  $syn = e$  pour les neurones excitateurs,  $i$  pour les neurones inhibiteurs et, lorsqu’un potentiel d’action se produit,

$$g_{syn} \rightarrow g_{syn} + w_{syn}$$

avec  $\tau_i = 10\text{ ms}$ ,  $\tau_e = 5\text{ ms}$ ,  $E_i = -80\text{ mV}$ ,  $E_e = 0\text{ mV}$ .  $w_{syn}$  représente le poids synaptique ou valeur d’incrément de la conductance synaptique.

### Quelques définitions

- Dans la suite, le terme “délai” désignera un délai de propagation entre deux neurones connectés. la plupart des simulations ont été effectuées sans délai, excepté dans la section 4.3.3.
- La “probabilité de connexion”,  $sp$ , dans un réseau, correspond ici au nombre moyen de connexions par neurone ; par exemple,  $sp = 0.5$  signifie que dans un réseau de 100 neurones, chacun est en moyenne connecté à 50 autres cellules. Autrement dit, de manière générale, l’espérance mathématique du nombre de connexions par neurone ( $\mathbf{E}(N_{connex/neur})$ ) est  $sp$  multiplié par le nombre de neurones du réseau ( $sp \times N$ ).

### Méthodes

#### – Logiciel et méthode d’intégration

Le logiciel de simulation *Brian* développé par R. Brette et D. Goodman [GB08] a été choisi pour réaliser les simulations.

Nous avons résolu numériquement les équations du modèle en intégrant les équations différentielles grâce à la méthode d’Euler avec un pas de temps de  $0.1\text{ ms}$ , mais nous avons également testé l’implémentation avec un pas de temps plus petit ( $0.05\text{ ms}$ ) sans observer

de différence notable.

– **Bruit et hétérogénéité**

Du bruit et de l'hétérogénéité ont parfois été introduits dans les simulations ; ils s'exprimaient alors de la façon suivante :

– Stimulations bruitées :

$$I_{\text{app}} = I_{\text{constant}} + \mathcal{U}[-0.005; 0.005]nA \quad (4.4)$$

où  $I_{\text{constant}}$  est un courant constant généralement compris entre 0.5 et 2.5 nA.  $\mathcal{U}[-0.005; 0.005]$  désigne une valeurs tirée aléatoirement de manière uniforme dans l'intervalle  $[-0.005; 0.005]$  à chaque pas de temps et pour chaque neurone.

– Hétérogénéité :

l'étape suivante de cette expérimentation numérique étant d'implémenter les réseaux sur nos circuits neuromimétiques, sujets à une certaine variabilité des composants électroniques, nous avons introduit de l'hétérogénéité (perturbation paramétrique) par le biais du paramètre  $a$  en tirant aléatoirement et de manière uniforme sa valeur de la façon suivante :

$$a = (a_{\text{fix}} + \mathcal{U}[-0.01; 0.01]).g_L \quad (4.5)$$

où  $a_{\text{fix}}$  est une constante.

Cependant, une manière plus naturelle d'introduire de l'hétérogénéité aurait été de perturber aléatoirement  $\Delta_T$  ou  $g_L$ , les densités de canaux ioniques ou la résistance d'entrée. Des résultats similaires peuvent alors être obtenus avec ce type d'hétérogénéité (non démontré ici).

– **Un calcul de la cohérence dans un réseau de neurones**

Il existe plusieurs moyens de calculer la cohérence de l'activité d'un réseau de neurones. Nous avons, pour cette étude, retenu la mesure de cohérence,  $\kappa$ , décrite dans [WB96], calculée à partir de corrélations croisées de paires de neurones. La cohérence entre deux neurones  $x$  et  $y$  s'exprime alors en fonction des listes de potentiels d'actions correspondantes,  $X$  et  $Y$ , après avoir choisi la résolution temporelle,  $\tau_{\text{bin}}$  sur un intervalle  $T$  :

$$\kappa_{xy}(\tau_{\text{bin}}) = \frac{\sum_{l=1}^K X(l)Y(l)}{\sqrt{\sum_{l=1}^K X(l) \sum_{l=1}^K Y(l)}}$$

La liste des spikes est construite ainsi :  $X(l) = 1$  (s'il y au moins un spike dans la fenêtre de taille  $\tau_{\text{bin}}$ ) ou 0 sinon et de même,  $Y(l) = 0$  ou 1,  $l = 1, 2, \dots, K$  ( $K = \tau_{\text{bin}}/T$ ).

La mesure de cohérence pour toute la population,  $\kappa(\tau_{\text{bin}})$ , que nous utiliserons plus loin, est calculée en moyennant les mesures de cohérence  $\kappa_{xy}(\tau_{\text{bin}})$  de plusieurs paires de neurones tirées aléatoirement dans le réseau. Autrement dit, lorsque  $\tau_{\text{bin}}$  est très petit, l'activité du réseau est d'autant plus synchronisée que  $\kappa(\tau_{\text{bin}})$  est proche de 1 et moins le réseau est synchronisé, plus  $\kappa(\tau_{\text{bin}})$  est proche de 0.

## 4.3 Les différents réseaux de neurones envisagés

### 4.3.1 Réseau de neurones excitateurs et inhibiteurs de type I ou II exclusivement

Le premier réseau testé était constitué de 20 neurones excitateurs et 80 inhibiteurs avec les poids synaptiques suivants :  $w_e = 4,50 nS$  et  $w_i = 9,0 nS$ . Les connexions ont été initialisées aléatoirement avec une probabilité de connexion fixée pour chacun des deux groupes de neurones : la population excitatrice était connectée au reste du réseau (y compris les neurones excitateurs) avec une probabilité égale à 0.6 et la population inhibitrice avec une probabilité de 0.2. Il s'agissait de créer un réseau comportant un petit groupe de neurones excitateurs pour maintenir l'activité dans un réseau majoritairement constitué de cellules inhibitrices. Nous avons ensuite inversé les proportions de chaque catégorie afin de reproduire un réseau plus réaliste biologiquement (voir [BC95] ou [HS05]) ; le nouveau réseau fut alors constitué de 25 inhibiteurs et 75 excitateurs avec  $w_e = 3.50 nS$  et  $w_i = 7.0 nS$ , où les connexions furent initialisées aléatoirement avec une probabilité de connexion de 0.6 pour les connexions inhibitrices et 0.3 pour les connexions excitatrices. La figure 4.2 résume ces informations.

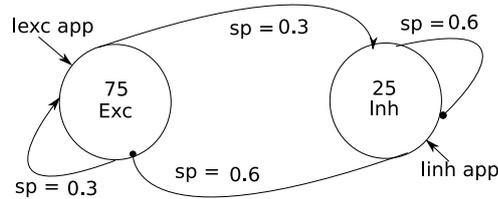


FIG. 4.2 – Schéma de connexion du réseau de neurones.

### Avec des neurones résonateurs de type II

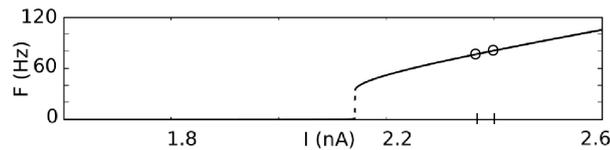


FIG. 4.3 – Courbe donnant la fréquence d'oscillation d'un neurone résonateur de type II en fonction du courant appliqué. Les cercles correspondent aux valeurs des courants de stimulations choisis pour les simulations.

Les simulations ont été effectuées avec stimulations bruitées et non bruitées ainsi qu'avec et sans hétérogénéité. Lorsque le courant de stimulation,  $I_{app}$  n'était pas bruité, il était constant durant toute la simulation ; lorsque qu'au contraire il était perturbé aléatoirement, il s'exprimait comme décrit dans la sous-section précédente.

Nous avons choisi les valeurs des paramètres comme suit :

$$a = 3g_L \text{ et } \tau_w = 2\tau_m, \text{ et lorsqu'il y avait hétérogénéité : } a_{fix} = 3.$$

Les autres paramètres restent inchangés.

Ces neurones sont des résonateurs de type II, ainsi, lorsque nous considérons un neurone, celui-ci oscille pour toutes ou presque toutes les valeurs de courant de stimulation  $I_{app}$ . La stimulation des neurones excitateurs,  $I_{exc, app}$ , doit être suffisamment élevée pour permettre la génération de potentiels d'action, mais le réseau nécessite la présence d'interactions inhibitrices pour éviter des épisodes de bursts trop longs ; il faut donc stimuler également les neurones inhibiteurs afin de leur permettre de générer des potentiels d'action en présence des connexions excitatrices. Cependant, il est évident que la stimulation imposée aux neurones inhibiteurs ne doit pas être trop forte, auquel cas, l'activité de tout le réseau serait inhibée par celle des neurones inhibiteurs. Ces mêmes observations sont valables pour les réseaux que nous décrirons plus loin (type I uniquement et réseau mixte), nous ne les rappellerons donc pas dans les prochaines sections.

Lorsque les neurones inhibiteurs sont stimulés en deçà du seuil de déclenchement du potentiel d'action – seuil pour une cellule isolée –, la stimulation appliquée aux neurones excitateurs impose le rythme d'oscillation pour tout le réseau, en particulier la fréquence des oscillations gamma lorsqu'elles apparaissent ; les connexions inhibitrices, quant à elles, ajoutent une certaine hétérogénéité dans l'activité et permettent au réseau de se synchroniser en modérant son activité. Afin de pouvoir observer des oscillations gamma, nous avons noté qu'il était nécessaire que le courant de stimulation appliqué à la population excitatrice soit supérieur à  $2nA$  et que le courant imposé aux neurones inhibiteurs soit inférieur à  $I_{exc, app}$ , mais d'autant plus proche de celui-ci que  $I_{exc, app}$  est grand. Pour résumer, pour observer des oscillations gamma, il est plus facile de faire varier  $I_{inh, app}$  pour une valeur fixée de  $I_{exc, app}$  que de fixer  $I_{inh, app}$  et ajuster ensuite  $I_{exc, app}$ .

De plus, les oscillations observées lorsque le réseau se synchronise sont dans la bande de fréquence des oscillations gamma. La figure 4.4<sup>38</sup> montre un exemple de cette activité avec une stimulation bruitée et sans hétérogénéité et la figure 4.5, un exemple de l'activité du réseau avec une stimulation bruitée et une initialisation aléatoire de la tension de membrane, sans hétérogénéité ; les deux simulations ont été effectuées avec les mêmes valeurs de  $I_{constant}$ . Dans le deuxième cas, ce type de motif avec un temps d'“adaptation” se produit environ une fois sur quatre et apparaît plus fréquemment avec un réseau de 80 neurones inhibiteurs et 20 excitateurs (non prouvé ici). Aussi, quand l'activité se stabilise-t-elle, la fréquence d'oscillation du motif est de 28 Hz environ.

Quant à la synchronisation, celle-ci est extrêmement robuste et se maintient malgré des stimulations bruitées. Ce comportement est très similaire à ceux observés dans les réseaux de neurones excitateurs et inhibiteurs dits « feedforward » dans les couches cérébrales profondes (L6 to L11) du cortex du rat [Rey03]. Néanmoins, ce phénomène de synchronisation peut être “cassé” en ajoutant de l'hétérogénéité à chaque cellule suivant l'équation (eq.4.5) et en stimulant à la fois les neurones excitateurs et les neurones inhibiteurs au dessus du seuil individuel de déclenchement d'un potentiel d'action (les résultats sont alors similaires à ceux observés figure 4.8). Dans ce cas, il est aussi possible d'ajuster la fréquence d'oscillation du motif dans le réseau.

Des résultats similaires ont été observés avec le premier réseau testé (20 excitateurs, 80 inhibiteurs).

### Avec des neurones intégrateurs ou résonateurs de type I uniquement

Les simulations ont tout d'abord été effectuées avec 100 neurones intégrateurs de type I (et classe A) : haut de la figure fig. 4.6), puis avec 100 neurones résonateurs (classe A) (bas de la

<sup>38</sup>Le champ synaptique moyen est calculé de la façon suivante :  $g_{e,mean} = \sum_{k=1, \dots, N_{exc}} g_e(t, k) / (0,3N_{exc}(N_{exc} + N_{inh}))$  and  $g_{i,mean} = \sum_{k=1, \dots, N_{inh}} g_i(t, k) / (0,6N_{inh}(N_{exc} + N_{inh}))$  (nous divisons par la probabilité de connexion  $\times$  nombre maximal de connexions possibles).

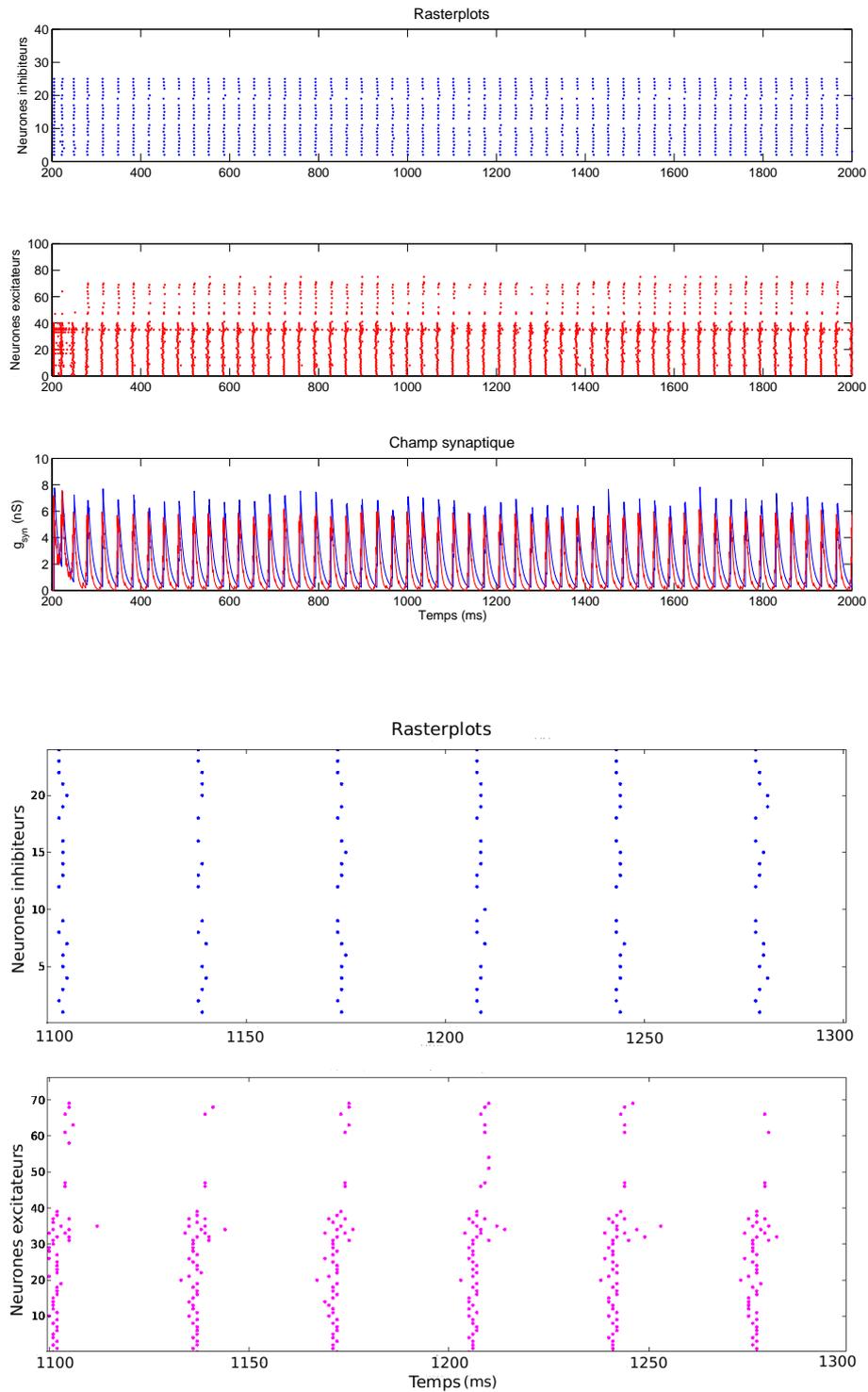


FIG. 4.4 – Réseau de type II – Rasterplots des deux populations avec  $I_{\text{constant\_inh}} = 1,4 nA$ ,  $I_{\text{constant\_exc}} = 2,4 nA$  pour les neurones 1 à 40 et  $I_{\text{constant\_exc}} = 0,6 nA$  pour les neurones 41 to 75. Haut : rasterplots et champ synaptique moyen pour 2 s de simulation. Bas : zoom autour de quelques événements.

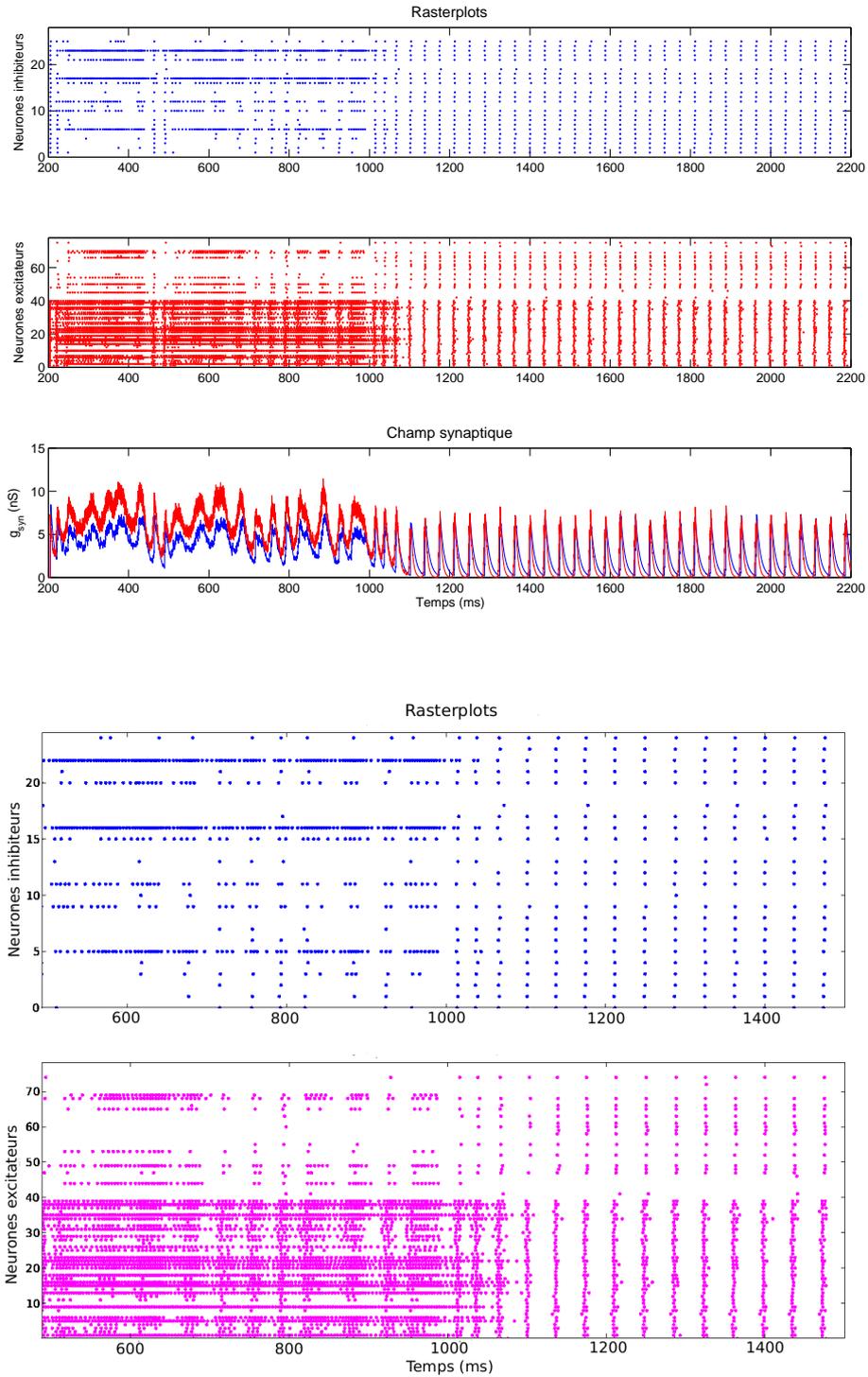


FIG. 4.5 – Réseau de type II – Rasterplots des deux populations avec  $I_{\text{constant\_inh}} = 1,4 nA$ ,  $I_{\text{constant\_exc}} = 2,4 nA$  pour les neurones 1 à 40 et  $I_{\text{constant\_exc}} = 0,6 nA$  pour les neurones 41 à 75. Haut : rasterplots et champs synaptiques moyen pour 2 s de simulation. Bas : zoom autour de quelques événements.

figure fig. 4.6). D'après [TB08] et le diagramme de la figure fig. 4.1, nous avons choisi dans le premier cas :

$$a = 0,2g_L \text{ et } \tau_w = \tau_m/3, \text{ et lorsque l'on ajoutait de l'hétérogénéité : } a_{\text{fix}} = 0,2;$$

et dans le second cas :

$$a = 0,6g_L \text{ et } \tau_w = \tau_m/2,5, \text{ et lorsque l'on ajoutait de l'hétérogénéité : } a_{\text{fix}} = 0,6.$$

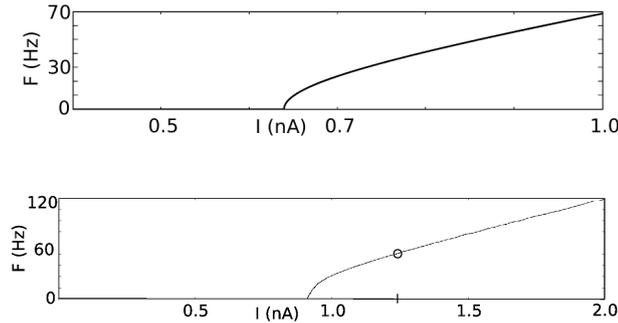


FIG. 4.6 – Courbes donnant la fréquence d’oscillation d’un neurone en fonction courant appliqué pour nos neurones de type I (et classe A) intégrateur (haut) et classe A résonateur (bas). Le cercle correspond à une valeur choisie pour la stimulation appliquée au réseau.

Dans les deux cas, comme avec le précédent réseau, la condition suivante devait être satisfaite pour observer des oscillations (presque) synchronisées :  $I_{\text{exc, app}} > I_{\text{inh, app}}$ .

Avec ce type de neurones, la fréquence des oscillations peut être abaissée au-dessous de la bande de fréquence gamma (jusqu’à 2 Hz ou 1 Hz) en diminuant la valeur de  $I_{\text{exc, app}}$ , alors que la fréquence d’oscillation du motif pour le réseau de neurones de type II ne peut pas être abaissée en-dessous de 18 Hz environ. La figure 4.7 illustre le fonctionnement du réseau à basse fréquence, avec un motif se répétant à 2,5 Hz et sur les figures 4.8 à 4.10, les deux populations oscillent dans la bande de fréquence gamma (autour de 40 Hz).

Concernant la synchronisation, la seule différence observée entre le réseau d’intégrateurs et le réseau de résonateurs est que la synchronisation est plus robuste avec les résonateurs qu’avec les intégrateurs. En effet, pour “rompre” la synchronisation, il était nécessaire de rendre plus hétérogène le réseau (en terme de paramètres du modèle pour chaque cellule). Cependant, dans ces conditions, les oscillations du réseau et la synchronisation (partielle) étaient alors plus sensibles à la connectivité; en d’autres termes, si la connectivité était “mauvaise<sup>39</sup>”, le réseau n’ exhibait pas de motif d’oscillations.

Dans les deux cas, lorsque les oscillations apparaissent et lorsque les neurones inhibiteurs ne sont pas des oscillateurs (ie, lorsqu’ils sont stimulés en-dessous du seuil individuel de déclenchement d’un potentiel d’action), le choix de la valeur de  $I_{\text{exc, applied}}$  permet d’ajuster la fréquence d’oscillation du réseau.

Finalement, ce type de réseau semble moins sensible à une initialisation aléatoire des valeurs de  $V_{\text{mem}}$  que les réseaux de neurones de type II : aucun “temps d’adaptation” n’a été observé lorsque l’on initialisait la valeur de la tension de membrane de chaque neurone de manière aléatoire.

<sup>39</sup>La connectivité requise n’a pas été étudiée ici.

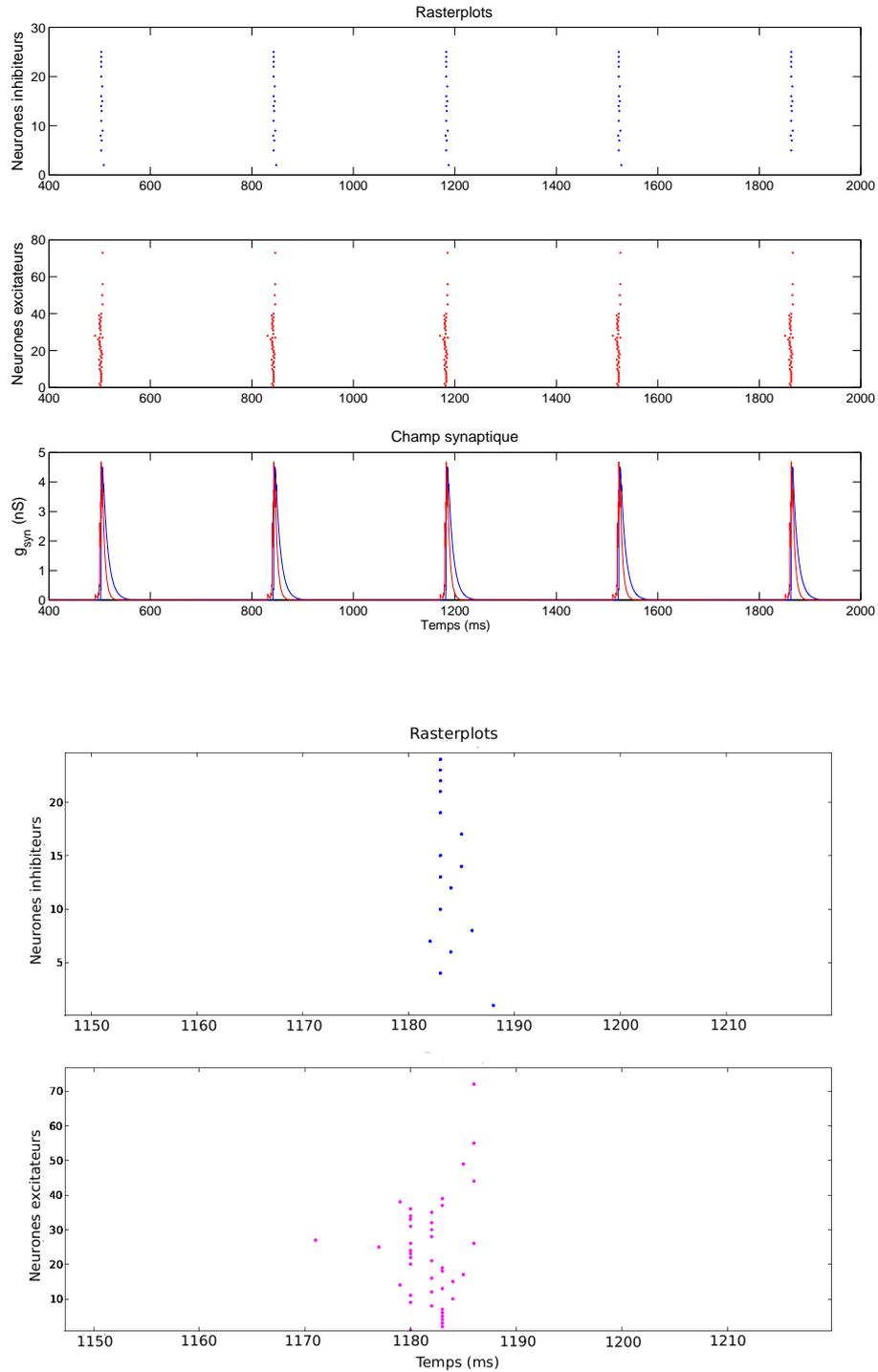


FIG. 4.7 – Rasterplots des populations avec  $I_{\text{constant\_inh}} = 1,2 \text{ nA}$ ,  $I_{\text{constant\_exc}} = 1,3 \text{ nA}$  pour les neurones 1 à 40 et  $I_{\text{constant\_exc}} = 0,8 \text{ nA}$  pour les neurones 41 à 75. Les neurones sont des intégrateurs. Haut : rasterplots et champ synaptique moyen pour 1,5 s de simulation. Bas : zoom autour d'un événement.

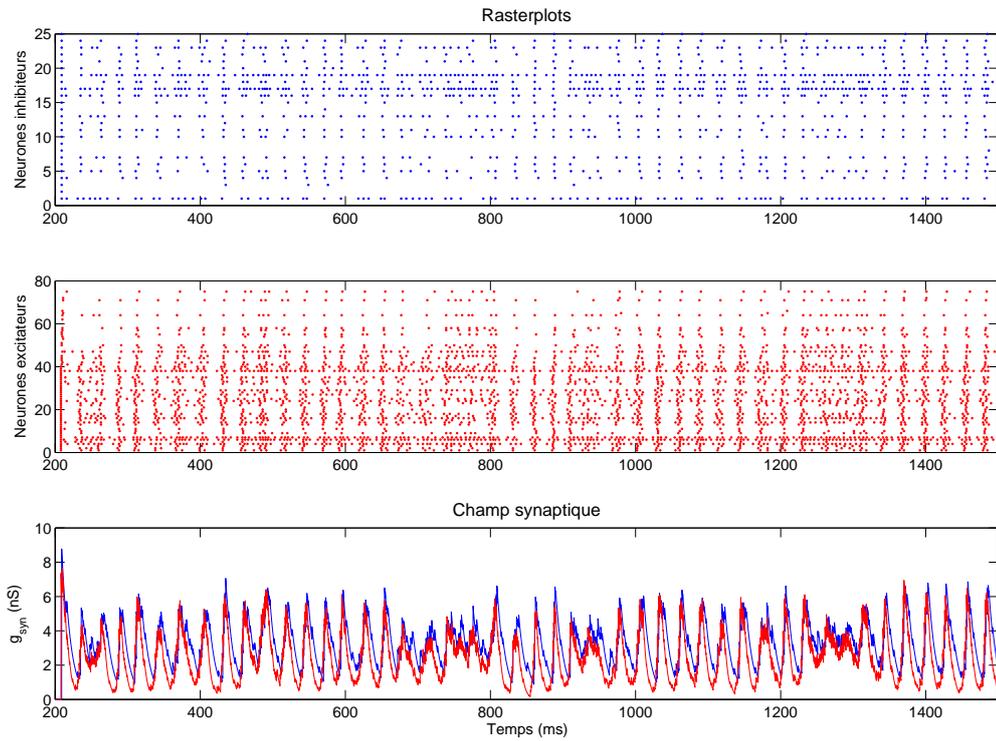


FIG. 4.8 – Réseau de type I – Rasterplots des populations avec  $I_{\text{constant\_inh}} = 1,1 \text{ nA}$ ,  $I_{\text{constant\_exc}} = 1,3 \text{ nA}$  pour les neurones 1 à 40 et  $I_{\text{constant\_exc}} = 0,8 \text{ nA}$  pour les neurones 41 à 75. Les neurones sont des intégrateurs.

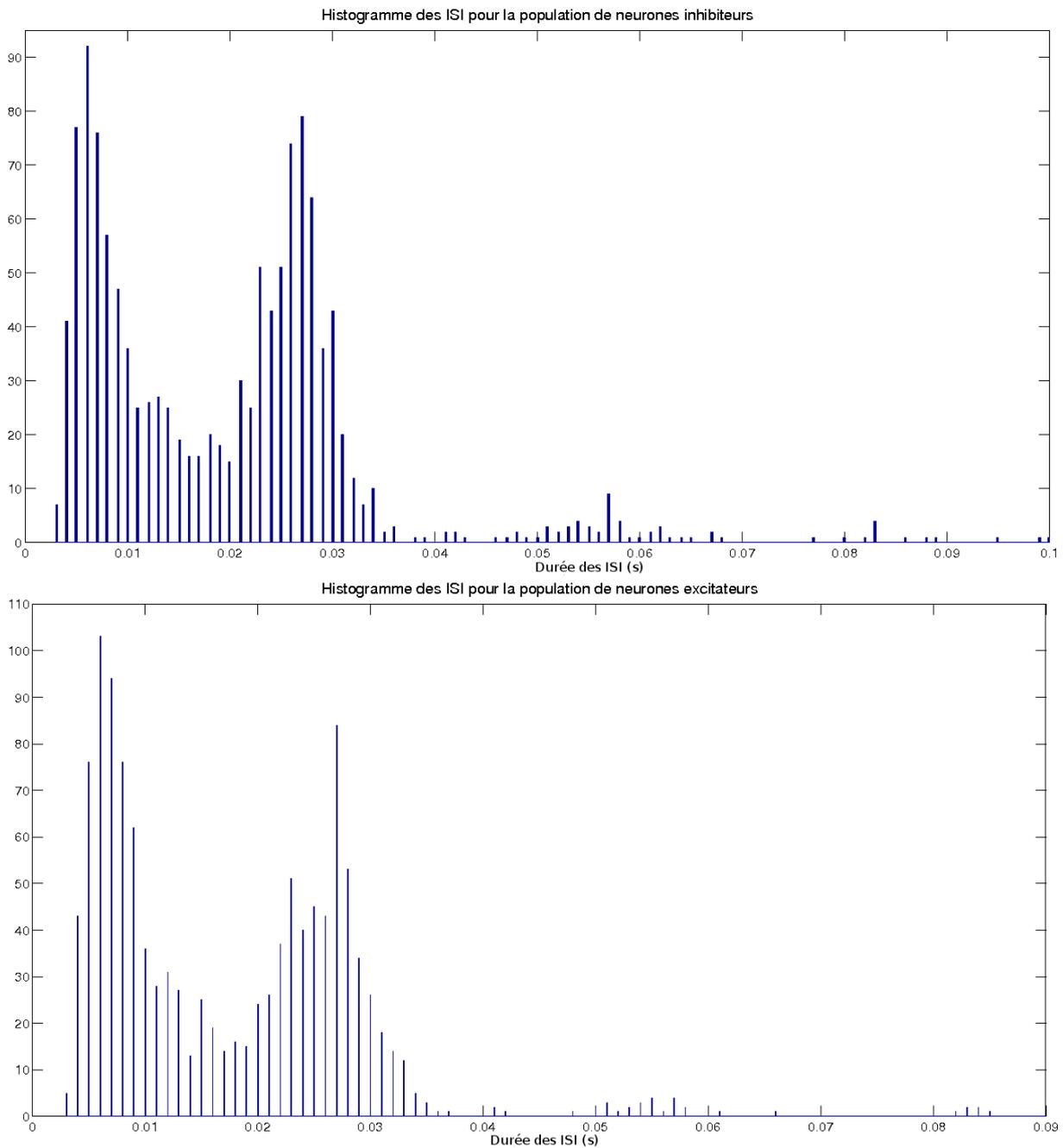


FIG. 4.9 – Réseau de type I – Histogramme des Intervalles InterSpike (ISI) pour les deux populations (avec  $I_{\text{constant\_inh}} = 1,1 nA$ ,  $I_{\text{constant\_exc}} = 1,3 nA$  pour les neurones 1 à 40 et  $I_{\text{constant\_exc}} = 0,8 nA$  pour les neurones 41 à 75. Les neurones sont des intégrateurs.

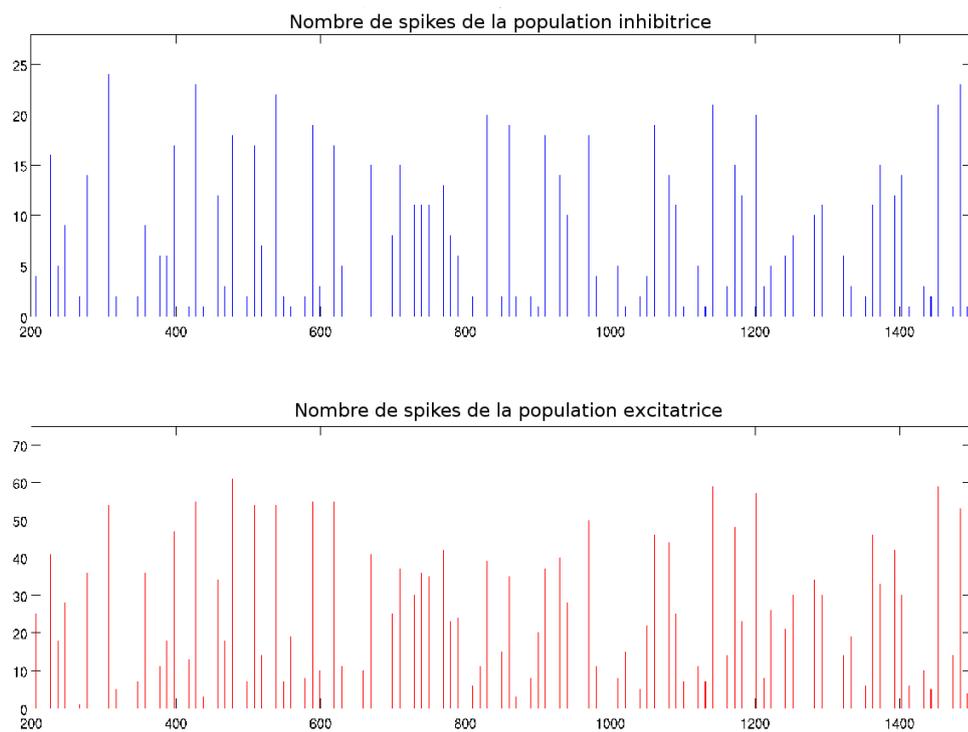


FIG. 4.10 – Réseau de type I – Nombre de spikes dans des fenêtres de  $10\text{ ms}$  (avec  $I_{\text{constant\_inh}} = 1,1\text{ nA}$ ,  $I_{\text{constant\_exc}} = 1,3\text{ nA}$  pour les neurones 1 à 40 et  $I_{\text{constant\_exc}} = 0,8\text{ nA}$  pour les neurones 41 à 75). Les neurones sont des intégrateurs.

### 4.3.2 Réseau mixte de neurones excitateurs de type I et d'inhibiteurs de type II

Pour cette configuration, les neurones inhibiteurs ont été implémentés avec le modèle de type II utilisé précédemment ; nous avons ensuite réalisé deux types de simulations, les premières avec des neurones excitateurs de type I intégrateurs, les secondes avec des résonateurs.

Nous pouvons d'ores et déjà faire la même observation que dans la section précédente : la synchronisation est plus robuste avec des neurones excitateurs de type résonateurs qu'avec des intégrateurs, et comme les neurones excitateurs déterminent la vitesse d'oscillation du réseau, la fréquence du motif – lorsqu'il y a synchronisation – peut être abaissée en-dessous de la plage de fréquence gamma. Or, d'après Ermentrout, un réseau de cellules uniquement excitatrices de type I ne peut pas se synchroniser à basses fréquences. Le résultat ci-dessus est donc intéressant et prouve l'importance de l'inhibition. Il serait intéressant d'effectuer des mesures sur tranche afin d'observer si, en diminuant la stimulation, on parvient à faire diminuer la fréquence d'apparition du motif, auquel cas nous pourrions faire des prédictions, à partir de notre modèle, sur la composition de tranche étudiée : elle comporterait bien des neurones excitateurs de type I et des neurones de type II si la fréquence d'oscillation pouvait être abaissée.

Les figures suivantes fournissent un exemple du comportement du réseau dans la bande gamma avec des neurones excitateurs résonateurs : la figure 4.11 représente le rasterplot des deux populations. On peut voir, grâce à la figure 4.12, que la fréquence du motif est d'environ 30 Hz (second pic dans le diagramme) avec quelques “bursts” (125 Hz, premier pic) ; quelques neurones déchargent moins de deux fois pendant les 1000 ms de simulation (ils n'apparaissent pas dans le diagramme pour des raisons de visibilité de la figure). Le nombre de spikes par unités de temps (ici, 10 ms) apparaît sur la figure 4.13 pour la première seconde de simulation : ceci montre par exemple que l'activité cesse autour de 750 ms car un grand nombre de neurones inhibiteurs – en comparaison du nombre de neurones excitateurs – sont actifs au même moment (rectangles noirs sur la figure).

De plus, la cohérence du réseau dépend du nombre de neurones excitateurs spontanément actifs (que nous appellerons aussi “neurones excitateurs oscillateurs”). Nous dirons que des neurones sont spontanément actifs s'ils sont individuellement stimulés au-dessus de leur seuil de déclenchement, ie si cette stimulation leur permet d'émettre des potentiels d'action en l'absence de connexions. La figure 4.14 fournit la mesure de cohérence en fonction du nombre de neurones excitateurs oscillateurs pour une stimulation moyenne fixée et correspondant à un motif d'oscillations de 30 Hz. Des résultats comparables ont été observés pour d'autres fréquences d'oscillation et nous avons noté également que le nombre d'oscillateurs ne semblait pas affecter la fréquence. Pour ces simulations, tous les neurones inhibiteurs étaient des oscillateurs. Lorsque le nombre de neurones non oscillateurs dépasse 30, peu, parmi ceux-ci, émettent des potentiels d'action ; néanmoins, les motifs “presque synchronisés” apparaissent tant que le nombre de non-oscillateurs ne dépasse pas 50 (sur les 75 cellules excitatrices), au delà, l'activité se “désynchronise”. Cependant, moins il y a d'oscillateurs dans le réseau, moins les motifs sont reconnaissables : on peut observer ce phénomène sur la figure 4.15 où est tracé le ratio entre  $\kappa(5)$  et  $\kappa(10)$ . En effet, plus ce ratio est proche de 1, plus le motif est visible car la plupart des potentiels d'action sont émis dans une fenêtre temporelle courte de 5 ms et peu de potentiels d'action supplémentaires apparaissent dans une fenêtre temporelle de 10 ms.

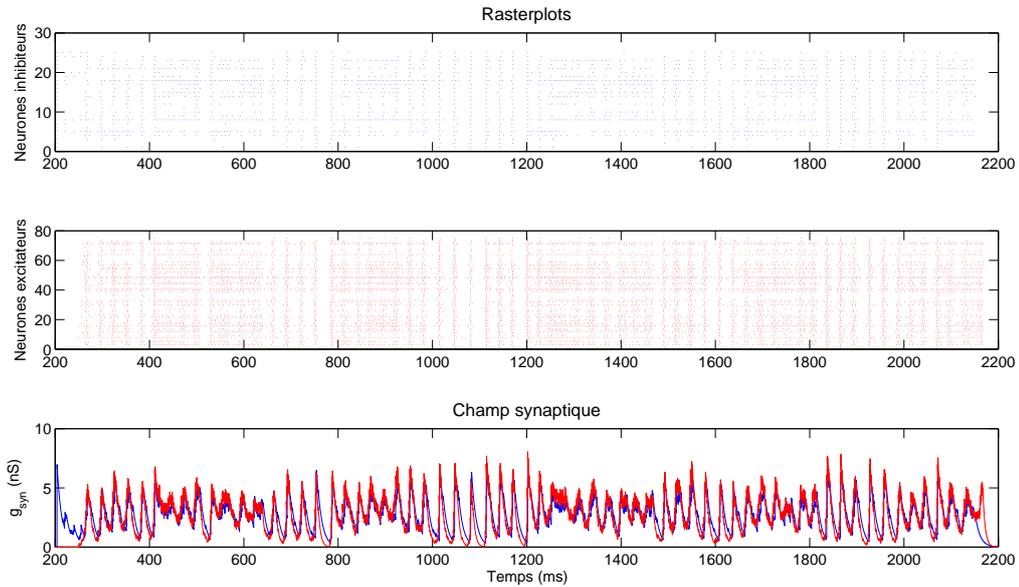


FIG. 4.11 – Réseau mixte type I/type II – Rasterplots des deux populations avec  $I_{\text{constant\_inh}} = 2.38 \text{ nA}$  (voir fig. 4.3) et  $I_{\text{constant\_exc}} = 1.25 \text{ nA}$  (voir fig. 4.6). Les neurones excitateurs sont des résonateurs.

### 4.3.3 Réseau de neurones inhibiteurs

Le réseau est constitué de 100 neurones de type II avec les mêmes paramètres que ceux utilisés dans la section 4.2.2, avec de l'hétérogénéité :  $a_{\text{fix}} = 3$  et  $\tau_w = 2\tau_m$ . D'après les observations faites dans [WB96], nous avons choisi une probabilité de connexion égale à 0.8 (ie 80 synapses par neurone en moyenne) de sorte qu'il y ait au moins 60 connexions par neurone afin de permettre au réseau de se synchroniser, puis nous avons ajouté un court délai de propagation d' $1 \text{ ms}$  (comme suggéré dans [BVJ07]). Le bruit et l'hétérogénéité ont été générés de la même façon que dans la section précédente. Le  $I_{\text{constant}}$  de l'équation eq. 4.4 était le même pour tous les neurones. Malgré tout, l'activité du réseau restait très synchronisée (la mesure de cohérence pour une fenêtre de  $5 \text{ ms}$  était proche de 0,85). On pourrait donc s'interroger sur l'influence du modèle choisi pour les synapses : observerait-on le même mécanisme de synchronisation si nous remplaçons les synapses exponentielles par un modèle bi-exponentiel (croissance et décroissance exponentielle) au lieu d'ajouter un délai de propagation ?

La figure 4.16 permet de comparer la fréquence d'oscillation du réseau et la fréquence du neurone seul pour une stimulation moyenne  $I_{\text{constant}}$  fixée. Le fait que les neurones soient connectés fortement ralentit l'activité globale du réseau. Ce dernier phénomène a déjà été observé dans [KE04].

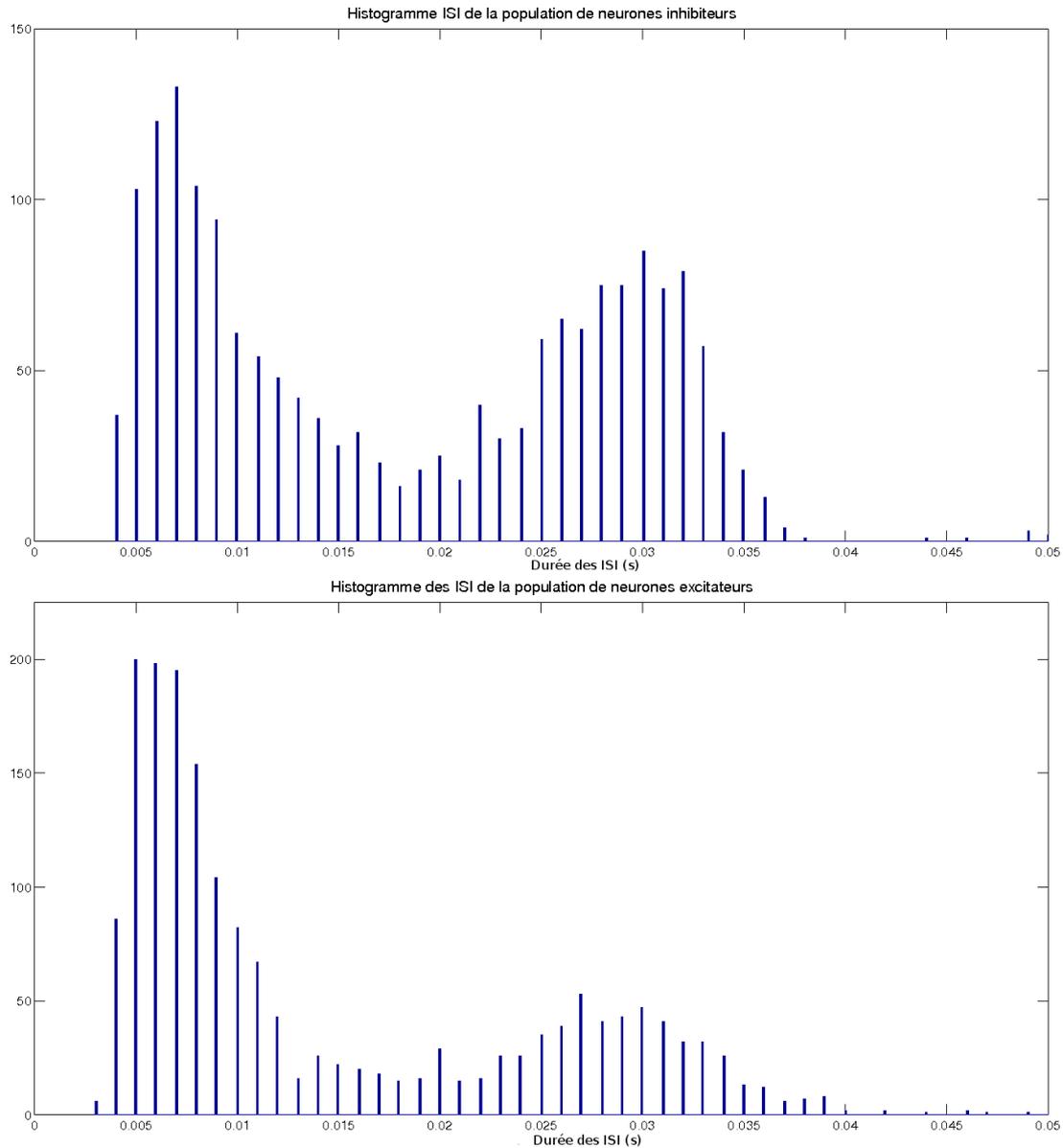


FIG. 4.12 – Réseau mixte type I/type II – Histogramme des Intervalles InterSpike (ISI) pour les deux populations (avec  $I_{\text{constant\_inh}} = 2.38 \text{ nA}$  (voir fig. 4.3) et  $I_{\text{constant\_exc}} = 1.25 \text{ nA}$  (voir fig. 4.6). Les neurones excitateurs sont des résonateurs.

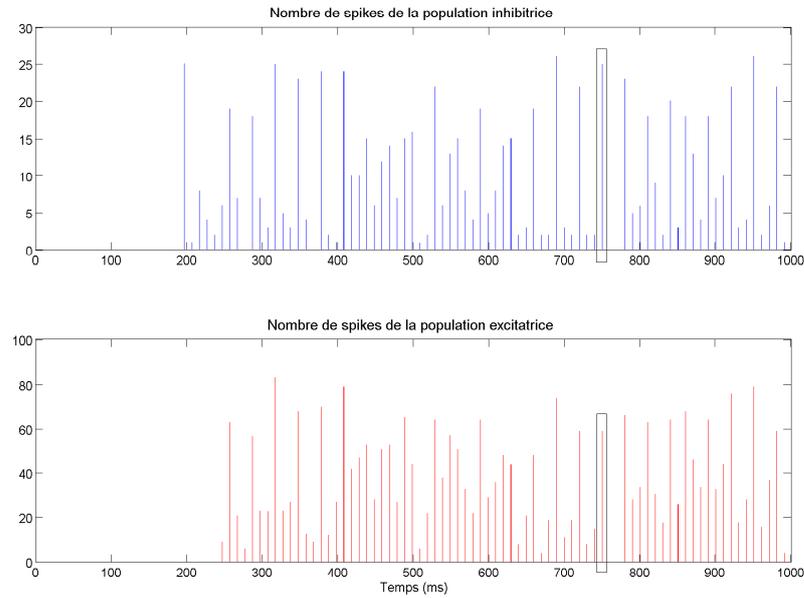


FIG. 4.13 – Réseau mixte type I/type II – Nombre de spikes dans des fenêtres de  $10\text{ ms}$  (avec  $I_{\text{constant\_inh}} = 2.38\text{ nA}$  (voir fig. 4.6) et  $I_{\text{constant\_exc}} = 1.25\text{ nA}$  (voir fig. 4.6)). Les neurones excitateurs sont des résonateurs.

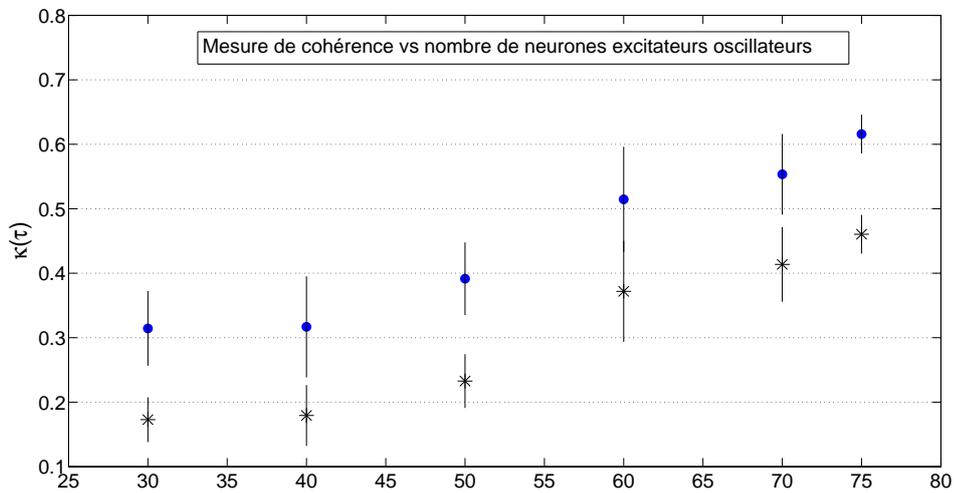


FIG. 4.14 – Réseau mixte type I/type II – Mesure de cohérence vs nombre de neurones excitateurs oscillateurs – \* avec  $\tau = 5\text{ ms}$ ,  $\cdot$  avec  $\tau = 10\text{ ms}$  –  $I_{\text{constant\_inh}} = 2.38\text{ nA}$  (voir fig. 4.3),  $I_{\text{constant\_exc}} = 1.25\text{ nA}$  (voir fig. 4.6) pour les oscillateurs et  $I_{\text{constant\_exc}} = 0.6\text{ nA}$  (voir fig. 4.6) pour les autres cellules.

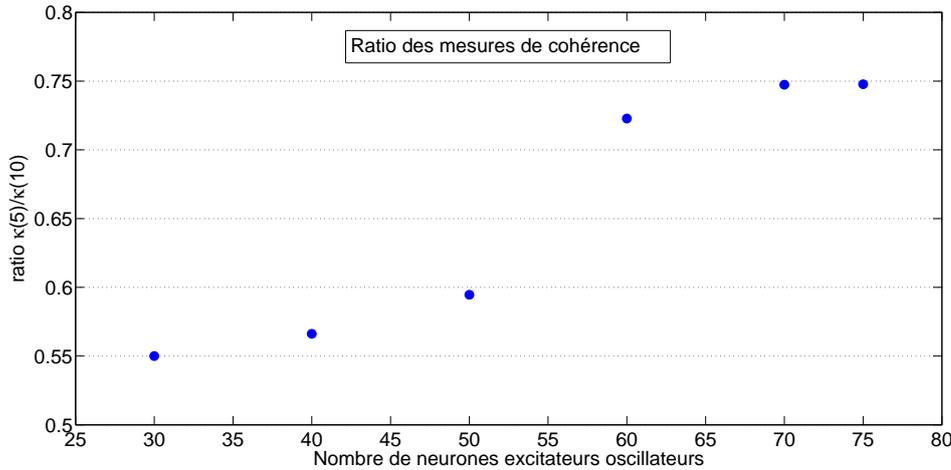


FIG. 4.15 – Réseau mixte type I/type II – Visibilité du motif : ratio des mesures de cohérence  $\kappa(5\text{ ms})$  et  $\kappa(10\text{ ms})$  –  $I_{\text{constant\_inh}} = 2.38\text{ nA}$  (voir fig. 4.3),  $I_{\text{constant\_exc}} = 1.25\text{ nA}$  (voir fig. 4.6) pour les oscillateurs et  $I_{\text{constant\_exc}} = 0.6\text{ nA}$  (voir fig. 4.6) pour les autres cellules.

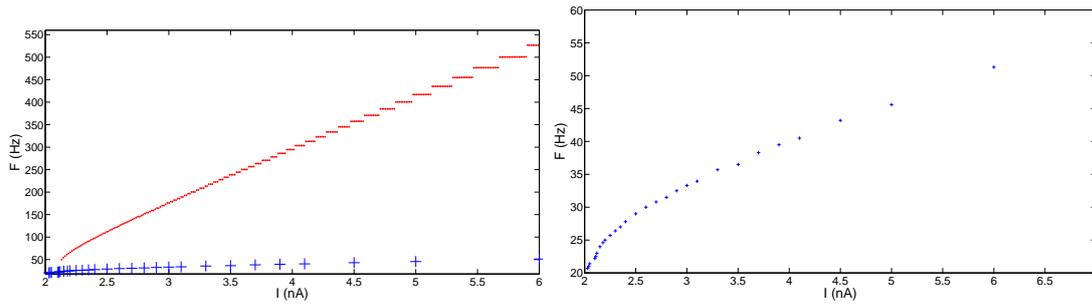


FIG. 4.16 – Réseau de neurones inhibiteurs – Fréquence vs  $I_{\text{constant}}$ . Gauche : “ + ” courbe bleue pour l’activité globale du réseau, “ . ” courbe rouge pour un neurone isolé. Droite : courbe pour l’activité globale du réseau.

## 4.4 Conclusion

Pour conclure sur cette partie, il est possible de générer des oscillations gamma à partir de réseaux comportant des neurones inhibiteurs et excitateurs, mais également de réseaux constitués uniquement de neurones inhibiteurs.

Nous avons observé que pour obtenir des oscillations (presque) synchronisées, les cellules excitatrices devaient être plus rapide (lorsqu’elles sont prises isolément) que les cellules inhibitrices. Autrement dit, dans un réseau de neurones de type I uniquement, ou de type II uniquement, la condition suivante doit être satisfaite :  $I_{\text{inh, app}} < I_{\text{exc, app}}$  ; mais cette condition peut ne pas être nécessaire et dépendre alors de la connectivité. En effet, si  $I_{\text{inh, app}}$  est plus grand que  $I_{\text{exc, app}}$ , l’activité du réseau ne peut généralement pas perdurer car l’influence inhibitrice est trop forte (les neurones inhibiteurs oscillent à une fréquence trop élevée). En revanche, si ces derniers sont insuffisamment stimulés, l’activité du réseau devient incontrôlable car aucun phénomène ne s’oppose à l’activité des neurones excitateurs. Aussi la synchronisation du réseau apparaît-elle lorsque le réseau est bien équilibré et permet à  $g_i$  et  $g_e$  de rester strictement positifs : après

une période de “silence” du réseau, l’activité peut réapparaître lorsque  $g_i$  devient suffisamment faible pour ne plus inhiber le reste du réseau et si  $g_e$  reste positif et peut entraîner à nouveau les neurones excitateurs.

Dans le cas des réseaux mixtes d’excitateurs et d’inhibiteurs, la dynamique du réseau dépend fortement des propriétés intrinsèques des cellules qui le constituent, en particulier de leur capacité d’oscillation à basses fréquences. Quand les cellules excitatrices sont de type I, nous observons avec notre modèle, que le réseau peut émettre des potentiels d’action synchronisés à très basses fréquences (inférieures à 2 Hz) ; à l’inverse, lorsque ces neurones sont de type II, il est impossible d’obtenir des motifs oscillant à des fréquences inférieures à 18 Hz. De plus, il semble que lorsque les neurones excitateurs sont des résonateurs, il soit plus difficile de “rompre” la synchronisation du motif et que la cohérence soit très élevée en dépit de l’ajout d’une hétérogénéité modérée ; lorsque que l’on accroît trop l’hétérogénéité, les motifs n’apparaissent plus. Dans ces réseaux mixtes, indépendamment du type de neurone, il est plus facile de rompre la synchronisation lorsque tous, excitateurs et inhibiteurs, sont des oscillateurs (ie sont stimulés au-dessus du seuil individuel de déclenchement d’un potentiel d’action) que lorsque seuls les neurones excitateurs sont des oscillateurs. La cohérence et la visibilité des motifs d’oscillation du réseau mélangeant types I et II augmentent quant à eux avec le nombre de neurones excitateurs spontanément actifs.

Concernant le réseau de neurones inhibiteurs, plusieurs questions nécessiteraient davantage d’investigation. Premièrement, la cohérence diminuerait-elle si, au lieu de n’implémenter que des cellules de type II, nous introduisons une fraction de cellules de type I ? Les premières simulations effectuées en ce sens suggèrent en effet que la cohérence décroît avec l’augmentation du nombre de neurones de type I. Deuxièmement, y aurait-il un moyen de prédire la fréquence d’oscillation du réseau connaissant les courants de stimulation et le champ synaptique ?

Finalement, nous avons pu constater qu’une synchronisation se produisait même en présence de stimulations bruitées et de neurones hétérogènes. L’implémentation de ces réseaux sur des circuits analogiques neuromimétiques, soumis à la dispersion des composants électronique et au bruit, nous permettrait donc de reproduire les activités gamma observées et d’approfondir l’étude présentée ici grâce aux simulations en temps réel que permet le système.

## 4.5 Résumé

Ce chapitre est un premier pas vers la configuration et l'utilisation d'un réseau de neurones électroniques. Après nous être intéressés à l'ajustement des paramètres d'une cellule individuellement, nous nous sommes orientés vers le réseau de neurones en effectuant une expérimentation numérique, par simulation logicielle, de l'influence des propriétés intrinsèques d'une cellule sur le comportement global d'un réseau de neurones dans le cadre des oscillations gamma. Cette étude a été réalisée avec la collaboration de John Rinzel, à l'Université de New York, au Courant Institute of Mathematical Science et au Center for Neural Science. La première section de ce chapitre décrit brièvement l'état de l'art en matière d'étude des oscillations gamma. La deuxième section se consacre aux travaux que j'ai effectués à New York : nous étudions dans cette section l'habilité du réseau à générer des oscillations gamma en fonction du type de neurones qui le constituent. Pour cela, nous proposons différents réseaux comportant chacun 100 neurones, d'abord inhibiteurs et excitateurs, tous de Type I ou bien tous de Type II ; ensuite en modélisant les neurones excitateurs par des modèles de Type I et les inhibiteurs par des modèles de Type II ; enfin, nous étudions un réseau de neurones inhibiteurs uniquement. Nous mettons ainsi en évidence la dépendance du comportement global du réseau vis à vis des caractéristiques individuelles des groupes de neurones qu'il comporte. Les techniques de réglage développées précédemment devrait permettre, dans un futur proche, de régler individuellement chaque cellule pour reproduire et approfondir cette dernière étude à l'aide des circuits neuromimétiques.

# Conclusion et perspectives

## Conclusion

Cette thèse, réalisée dans une équipe concevant des neurones sur silicium, avait pour objet l'estimation des paramètres de modèles de neurones biologiques à l'aide de techniques d'optimisation. Nous avons donc présenté, en premier chapitre de ce manuscrit, le lien étroit qui unissait la modélisation de neurones à diverses échelles – de la cellule au réseau –, à l'optimisation numérique, en tentant de définir au mieux chacune d'entre elle. Nous avons également introduit dans ce même chapitre le modèle d'Hodgkin-Huxley, un modèle de neurone biologique à l'échelle cellulaire, utilisé dans l'équipe pour la réalisation de circuits neuromimétiques. La présentation de ce modèle était accompagnée de la description de la méthode de voltage-clamp, technique neurophysiologique permettant d'enregistrer séparément les différents courants ioniques traversant la membrane du neurone, et de la méthode d'estimation associée servant à extraire les paramètres du modèle. Nous avons alors mis en évidence les limitations de cette méthode d'estimation et en avons développé, dans le deuxième chapitre, une alternative permettant d'estimer conjointement tous les paramètres d'un même canal ionique – à l'inverse de la méthode classique qui estimait un à un les paramètres par approximations successives. L'élaboration de cette nouvelle méthode a nécessité la comparaison de différents algorithmes d'optimisation, quatre métaheuristiques : le recuit simulé, les algorithmes génétiques, les essais particuliers et l'évolution différentielle. La dernière d'entre elles a retenu notre attention pour ses performances sur notre problème, en terme de vitesse de convergence et de qualité de la solution sur des données bruitées (issues d'enregistrements de courants provenant des circuits neuromimétiques) et non bruitées (validation logicielle). Nous avons également tenté d'améliorer l'algorithme d'évolution différentielle et prouvé l'efficacité des variantes sur un benchmark de fonctions. Le développement de la technique d'extraction de modèles de neurones a donné lieu à deux types d'applications présentées dans le chapitre 3 : l'estimation des paramètres du canal sodique du modèle d'Hodgkin-Huxley à partir d'enregistrements biologiques et le réglage automatique de circuits intégrés neuromimétiques. Le protocole automatisé d'ajustement des paramètres est maintenant utilisé pour régler les circuits de l'équipe en vue de constituer des réseaux de neurones électroniques et devrait servir également par la suite, d'un point de vue purement électronique, à évaluer et compenser la dispersion de caractéristiques des composants. En outre, le but des projets naissants de l'équipe étant d'étudier le comportement de petits réseaux (une centaine) de neurones en fonction de leurs structures (connectivité) et de la dynamique (excitateur, inhibiteur, type I, type II, etc) des neurones qui les constituent, nous présentons dans les deux derniers chapitres à la fois, une méthode d'estimation des paramètres de modèles de neurones prenant en compte directement la dynamique de la tension de membrane – sans utiliser les enregistrements de courants ionique – et une étude théorique préalable à l'implantation des réseaux sur silicium. Cette étude a été réalisée logiciellement et visait à explorer l'influence des propriétés intrinsèques de la cellule sur le comportement global du réseau dans le cadre des oscillations gamma (20-80Hz), souvent

décrites dans la littérature comme vecteur d'information dans certaines structures cérébrales. Nous avons ainsi mis en évidence le fait que la plage de fréquence d'oscillation du réseau – lorsqu'il y avait synchronisation partielle de l'activité – dépendait fortement de la nature des neurones qui le constituaient. Ainsi, lorsqu'une partie des cellules est de type I, le réseau peut osciller à très basse fréquence, même lorsqu'il contient également des cellules de type II. Par ailleurs la cohérence de l'activité dépend aussi du nombre de neurones spontanément actifs.

Cette dernière étude n'a pu être implémentée sur les circuits neuromimétiques avant la rédaction de mon manuscrit de thèse, mais sera traitée dans les futurs projets. Le développement du protocole de communication entre les cartes contenant les neurones et permettant des les relier en réseaux (avec synapses, plasticité, délais de propagations, etc) a fait l'objet d'une autre thèse soutenue dans l'équipe il y a peu.

Concernant l'estimation des paramètres de modèles neuronaux à partir d'enregistrements de la tension de membrane, les travaux n'ont pas été menés à terme mais seront poursuivis avec la nouvelle génération de circuits neuromimétiques conçus par l'équipe. Il serait d'ailleurs intéressant de se demander quel rôle pourrait jouer l'optimisation multi-objectif dans la résolution de ce problème complexe. Permettrait-elle de trouver des solutions de meilleure qualité en minimisant le temps de convergence ou bien ajouterait-elle de la complexité sans que, pour autant, les solutions trouvées ne répondent davantage à nos besoins ?

Enfin, ces travaux ont permis, au sein de l'équipe, de rendre exploitable l'outil puissant que représente le circuit neuromimétique pour la communauté neuroscientifique. En effet, sans réglage automatique des circuits, il était difficilement envisageable de les connecter en réseaux tout en contrôlant individuellement la dynamique de chaque neurone. Nous disposons dorénavant d'un système capable de reproduire en temps réel l'activité d'un réseau d'une centaine de neurones artificiels qui permettra des études irréalisables *in vitro*. Nous pourrions maîtriser à la fois le comportement de chaque cellule tout en observant la dynamique globale du réseau. De plus, bien "réglé" un neurone électronique peut également être vu comme un groupe de neurones, ce qui ouvre la perspective de simulations de plus grands réseaux de neurones (voire de structures cérébrales). Il sera donc possible de constituer une "bibliothèque" de comportements neuronaux en fonction de la structure du réseau, de ses constantes de temps et des propriétés intrinsèques des neurones.

# Bibliographie principale

- [AS06] P. ACHARD et E. De SCHUTTER : Complex parameter landscape for a complex neuron model. *PLoS Comput Biol*, 2(7):e94, 07 2006.
- [BC95] G. BUZSÁKI et J.J. CHROBAK : Temporal structure in spatially organized neuronal ensembles : a role for interneuronal networks. *Current Opinion in Neurobiology*, 5(4):504 – 510, 1995.
- [BG05] R. BRETTE et W. GERSTNER : Adaptive Exponential Integrate-and-Fire Model as an Effective Description of Neuronal Activity. *J Neurophysiol*, 94(5):3637–3642, 2005.
- [BM06] G. BARANAUSKAS et M. MARTINA : Sodium Currents Activate without a Hodgkin and Huxley-Type Delay in Central Mammalian Neurons. *J. Neurosci.*, 26(2):671–684, 2006.
- [BPV<sup>+</sup>08] P. BESSON, V. POPOVICI, J.-M. VESIN, J.-P. THIRAN et M. KUNT : Extraction of audio features specific to speech production for multimodal speaker detection. *IEEE Transactions on Multimedia*, 10:63–73, Janvier 2008.
- [BSG<sup>+</sup>09] L. BUHRY, S. SAÏGHI, A. GIREMUS, E. GRIVEL et S. RENAUD : Automated Tuning of Analog Neuromimetic Integrated Circuits. *In Proceedings of BioCAS 2009 Conference on Biomedical Circuits and Systems*, pages 13–16, Beijing Chine, November 2009.
- [BVJ07] M. BARTOS, I. VIDA et P. JONAS : Synaptic mechanisms of synchronized gamma oscillations in inhibitory interneuron networks. *Nature Rev. Neurosci.*, 8:45–54, Janvier 2007.
- [CB98] J.J. CHROBAK et G. BUZSÁKI : Gamma oscillations in the entorhinal cortex of the freely behaving rat. *J Neurosci.*, 18:388–398, Janvier 1998.
- [CJWB03] J. CSICSVARI, B. JAMIESON, K. D. WISE et G. BUZSÁKI : Mechanisms of gamma oscillations in the hippocampus of the behaving rat. *Neuron*, 37:311–322, Janvier 2003.
- [CK02] M. CLERC et J. KENNEDY : The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE transactions on evolutionary computation*, 6:58–73, 2002.
- [CM60] K. S. COLE et J. W. MOORE : Ionic current measurements in the squid giant axon membrane. *J. Gen. Physiol.*, 44:123–167, 1960.
- [CM04] B. CHEN et C.F. MARTIN : Fitzhugh-nagumo model and signal processing in the visual cortex of fly. *In Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 1, pages 591–595 Vol.1, Dec. 2004.

- [Col49] K. S. COLE : Dynamic electrical characteristics of the squid axon membrane. *Arch. Sci. Physiol.*, 3:253–258, 1949.
- [Col55] K. S. COLE : *Electrochemistry in biology ad medicine*, chapitre Ions, potentials and the nerve impulse, pages 121–140. T. Shedlovsky (ed.), new York, Wiley, 1955.
- [COS<sup>+</sup>02] A.R.R. CASTI, A. OMURTAG, A. SORNBORGER, E. KAPLAN., B. KNIGHT, J. VICTOR et L. SIROVICH : A population study of integrate-and-fire-or-burst neurons. *Neural Comput.*, 14(5):957–986, 2002.
- [CSHF08] D. CSERCSIK, G. SZEDERKÉNYI, K.M. HANGOS et I. FARKAS : Parameter estimation of hodgkin-huxley model of gnRH neurons. *In Proceedings of the 9th International Phd. workshop : Young Generation Viewpoint*, 2008.
- [DBG<sup>+</sup>07] S. DRUCKMANN, Y. BANITT, S. A. GIDON, F. SCHÄRMANN, H. MARKRAM et I. SEGEV : A novel multiple objective optimization framework for constraining conductance-based neuron models by experimental data. *Frontiers in Neuroscience*, 1:7–19, November 2007.
- [DH01] A. DESTEXHE et J. HUGUENARD : *Computational Neuroscience : Realistic Modeling for Experimentalists*, chapitre 5 : Which Formalism to Use for Modeling Voltage-Dependent Conductances, page 347. CRC Press LLC, 2001.
- [DOK02] S. DOI, Y. ONODA et S. KUMAGAI : Parameter estimation of various hodgkin-huxley-type neuronal models using a gradient-descent learning method. *In SICE 2002 : Proceedings of the 41st SICE Annual Conference*, pages 1685–1688, Washington, DC, USA, August 2002. IEEE Computer Society.
- [Dou] V. DOUENCE : *Circuits et systèmes de modélisation analogique de neurones biologiques*. Thèse de doctorat, University Bordeaux 1 – n 2324.
- [DS08] J. DRÉO et P. SIARRY : Métaheuristiques d’optimisation vues sous l’angle de l’échantillonnage de distribution. *Journal Européen des Systèmes Automatisés*, 42:9–30, 2008.
- [DSR93] K. DOYA, A. I. SELVERSTON et P. F. ROWAT : A Hodgkin-Huxley type neuron model that learns slow non-spike oscillations. *In Proceedings of NIPS 1993*, pages 566–573, 1993.
- [Fit55] R. FITZHUGH : Mathematical models of threshold phenomena in the nerve membrane. *Bull. Math. Biophysics*, 17:257–278, 1955.
- [Fit61] R. FITZHUGH : Impulses and physiological states in theoretical models of nerve membrane. *Biophysical J.*, 1:445–466, 1961.
- [FJ04] V. FEOKTISTOV et S. JANAQI : Generalization of the strategies in differential evolution. *In Parallel and Distributed Processing Symposium, 2004. 18th International*, pages 165–170, Washington, DC, USA, April 2004. IEEE Computer Society.
- [GB08] D. GOODMAN et R. BRETTE : Brian : a simulator for spiking neural networks in python. *Frontiers in Neuroinformatics*, 2, 2008.
- [GGMA01] M. GOLDMAN, J. GOLOWASCH, E. MARDER et L.F. ABBOTT : Global Structure, Robustness, and Modulation of Neuronal Models. *J. Neurosci.*, 21(14):5229–5238, 2001.

- 
- [GGN08] J. GROLLEAU, E. GRIVEL et M. NAJIM : Two ways to simulate rayleigh fading based on the sinusoidal stochastic model. *IEEE Signal Processing Letters*, 15:107–110, 2008.
- [GK07] M. GURKIEWICZ et A. KORNGREEN : A numerical approach to ion channel modelling using whole-cell voltage-clamp recordings and genetic algorithm. *PLoS Comput Biol*, 3(8):169, August 2007.
- [GPB05] W.C. GERKEN, L.K. PURVIS et R.J. BUTERA : Genetic algorithm for optimization and specification of a neuron model. *In Engineering in Medicine and Biology Society 27th Annual International Conference*, pages 4321–4323, Washington, DC, USA, 2005. IEEE Computer Society.
- [GS89] C.M. GRAY et W. SINGER : Stimulus-specific neuronal oscillations in orientation columns of cat visual cortex. *Proceedings of the National Academy of Sciences of the USA.*, 86(5):1698–1702, mars 1989.
- [GSA07] W. Van GEIT, E. De SCHUTTER et P. ACHARD : Neurofitter : A parameter tuning package for a wide range of electrophysiological neuron models. *Frontiers in Neuroinformatics*, 1:1–18, 2007.
- [GSA08] W. Van GEIT, E. De SCHUTTER et P. ACHARD : Automated neuron model optimization techniques : a review. *Biological Cybernetics*, 99:241–251, 2008.
- [HAP06] Q. HUYS, M. AHRENS et L. PANINSKI : Efficient estimation of detailed single-neuron models. *J. Neurophysiol.*, 96:872–890, 2006.
- [HH52a] A.L. HODGKIN et A.F. HUXLEY : The components of membrane conductance in the giant axon of loligo. *Jour. of Physiology*, 117:473–496, 1952.
- [HH52b] A.L. HODGKIN et A.F. HUXLEY : Currents carried by sodium and potassium ions through the membrane of the giant axon of loligo. *Jour. of Physiology*, 116:449–472, 1952.
- [HH52c] A.L. HODGKIN et A.F. HUXLEY : The dual effect of membrane potential on sodium conductance in the giant axon of loligo. *Jour. of Physiology*, 117:497–506, 1952.
- [HH52d] A.L. HODGKIN et A.F. HUXLEY : A quantitative description of membrane current and its application to conduction and excitation in nerve. *Jour. of Physiology*, 117:500–544, 1952.
- [HHK49] A. L. HODGKIN, A. F. HUXLEY et B. KATZ : Ionic currents underlying activity in the giant axon of the squid. *Archi. Sci. Physiology*, 3:129–150, 1949.
- [HHK52] A. L. HODGKIN, A. F. HUXLEY et B. KATZ : Measurements of current-voltage relations in the membrane of the giant axon of loligo. *Jour. of Physiology*, 116:424–448, 1952.
- [HK49] A.L. HODGKIN et B. KATZ : The effect of sodium ions on the electrical activity of the giant axon of the squid. *Jour. of Physiology*, 108:37–77, 1949.
- [HMLS07] D. HAUFLER, F. MORIN, J. C. LACAILLE et F. K. SKINNER : Parameter estimation in single-compartment neuron models using a synchronization-based method. *Neurocomput.*, 70(10-12):1605–1610, 2007.
- [Hol75] J.H. HOLLAND : *Adaptation in natural and artificial systems : an introductory analysis with applications to biology, control, and artificial intelligence*. Ann Arbor, MI : University of Michigan Press, 1975.

- [HR84] J. L. HINDMARSH et R. M. ROSE : A model of neuronal bursting using three coupled first order differential equations. *In Proceedings of the R. Soc. London, Ser. B*, volume 221, pages 87–102, 1984.
- [HS05] J.A. HENRIE et R. SHAPLEY : LFP Power Spectra in V1 Cortex : The Graded Effect of Stimulus Contrast. *J Neurophysiol*, 94(1):479–490, 2005.
- [Izh03] E. M. IZHIKEVICH : Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, 14:1569–1572, 2003.
- [Jon75] K. De JONG : *An analysis of the behaviour of a class of genetic adaptive systems*. Thèse de doctorat, University of Michigan., Michigan, 1975.
- [KE95] J. KENNEDY et R. EBERHART : Particle swarm optimization. *In Proceedings of the IEEE International Conference on Neural Networks (Perth, Australia)*, volume 4, pages 1942–1948, 1995.
- [KE04] N. KOPELL et B. ERMENTROUT : Chemical and electrical synapses perform complementary roles in the synchronization of interneuronal networks. *Proceedings of the National Academy of Sciences of the United States of America*, 101(43):15482–15487, 2004.
- [KGJV83] S. KIRKPATRICK, C. D. GELATT, JR. et M. P. VECCHI : Optimization by simulated annealing. *Science*, 220:671–680, 1983.
- [KH05] R.A KOENE et M.E. HASSELMO : An Integrate-and-fire Model of Prefrontal Cortex Neuronal Activity during Performance of Goal-directed Decision Making. *Cereb. Cortex*, 15(12):1964–1981, 2005.
- [KPK05] N. KEREN, N. PELED et A. KORNGREEN : Constraining compartmental models using multiple voltage-recordings and genetic algorithms. *J. Neurophysiol*, 94: 3730–3742, 2005.
- [KSJ00] E. R. KANDEL, J.H. SCHWARTZ et T.M. JESSELL : *Principles of Neural Science*. 4th ed. McGraw-Hill, New York, 2000.
- [Lap] L.E. LAPICQUE : Recherches quantitatives sur l’excitation électrique des nerfs traitée comme une polarisation.
- [LDK07] K. LIU, X. DU et L. KANG : *Advances in Computation and Intelligence*, chapitre Differential Evolution Algorithm Based on Simulated Annealing. Springer Berlin / Heidelberg, 2007.
- [LM01] G. LEMASSON et R. MAEX : *Computational Neuroscience : Realistic Modeling for Experimentalists*, chapitre 1 : Introduction to equation solving and parameter fitting. CRC Press, 2001.
- [LVRE02] V. LYAKHOV, A. VOL et B. RAIKHLIN-EISENKRAFT : Adaptive activity of neural structures a thermodynamic approach. *Medical hypotheses*, 59:289–296, 2002.
- [Mar49] G. MARMONT : Studies on the axon membrane ; a new method. *J. Cell. Physiol.*, 34:351–382, 1949.
- [MD91] M. MAHOWALD et R. DOUGLAS : A silicon neuron. *Nature*, 354:515 – 518, décembre 1991.
- [ML81] C. MORRIS et H. LECAR : Voltage oscillations in the barnacle giant muscle fiber. *Biophysical J.*, 35:193–213, 1981.

- 
- [MMB07] J. MODOLO, E. MOSEKILDEB et A. BEUTER : New insights offered by a computational model of deep brain stimulation. *Journal of Physiology – Paris*, 101:56–63, Janvier–Mai 2007.
- [MMCS00] J. L. MADDEN, Z. B. MILED, R. C. Y. CHIN et J. SCHILD : On parameter estimation for neuron models. In *BIBE '00 : Proceedings of the 1st IEEE International Symposium on Bioinformatics and Biomedical Engineering*, page 253, Washington, DC, USA, 2000. IEEE Computer Society.
- [NAY62] J. NAGUMO, S. ARIMOTO et S. YOSHIKAWA : An active pulse transmission line simulating nerve axon. In *Proceedings of Institute of Radio Engineers*, volume 50, pages 2061–2070. IEEE, 1962.
- [ODRR09] A.M. OSWALD, B. DOIRON, J. RINZEL et A.D. REYES : Spatial Profile and Differential Recruitment of GABAB Modulate Oscillatory Activity in Auditory Cortex. *J. Neurosci.*, 29(33):10321–10334, 2009.
- [PAFH05] D. PURVES, G.J. AUGUSTINE, D. FITZPATRICK et W.C. HALL : *Neurosciences*. De Boeck, 2005.
- [PBM03] A.A. PRINZ, C.P. BILLIMORIA et E. MARDER : Alternative to Hand-Tuning Conductance-Based Models : Construction and Analysis of Databases of Model Neurons. *J Neurophysiol*, 90(6):3998–4015, 2003.
- [PTRM<sup>+</sup>08] M. POSPISCHIL, M. TOLEDO-RODRIGUEZ, C. MONIER, Z. PIWKOWSKA, T. BAL, Y. FRÉGNAC, H. MARKRAM et A. DESTEXHE : Minimal hodgkin–huxley type models for different classes of cortical and thalamic neurons. *Biological Cybernetics*, 99:427–441, 2008.
- [RBD07] M.S. REID, E.A. BROWN et S.P. DEWEERTH : A parameter-space search algorithm tested on a hodgkin-huxley model. *Biologicals Cybernetics*, 96:625–634, 2007.
- [RC07] A. RANGAN et D. CAI : Fast numerical methods for simulating large-scale integrate-and-fire neuronal networks. *Journal of Computational Neuroscience*, 22(1):81–100, February 2007.
- [Rey03] A.D. REYES : Synchrony-dependent propagation of firing rate in iteratively constructed networks in vitro. *Nature Neuroscience*, 6:593–599, Mai 2003.
- [RGPB10] C. ROSSANT, D. GOODMAN, J. PLATKIEWICZ et R. BRETTE : Automatic fitting of spiking neuron models to electrophysiological recordings. *Frontiers in Neuroinformatics*, 4, 2010.
- [RLLMA93] S. RENAUD-LEMASSON, G. LE MASSON, E. MARDER et L.F. ABBOT : Hybrid circuits of interacting computer model and biological neurons. In *Advances in Neural Information Processing Systems 5, [NIPS Conference]*, pages 813–819, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.
- [RV02] J. RIGET et J. VESTERSTROEM : A diversity-guided particle swarm optimizer - the arps0, 2002.
- [Sai04] S. SAIGHI : *Circuits et systèmes de modélisation analogique de réseaux de neurones biologiques : application au développement d'outils pour les neurosciences computationnelles*. Thèse de doctorat, University Bordeaux 1 – n 2891, Novembre 2004.

- [SP97] R. STORN et K. PRICE : Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Jour of Global Optimization*, 11:341–359, 1997.
- [Ste67] R. B. STEIN : Some models of neuronal variability. *Biophysical Journal*, 7:37–68, 1967.
- [SVLM98] P. SHROETER, J. M. VESIN, T. LANGENBERGER et R. MEULI : Robust parameter estimation of intensity distributions for brain magnetic resonance images. *IEEE Trans. on Medical Imaging*, 17(2):172–186, 1998.
- [TB08] J. TOUBOUL et R. BRETTE : Dynamics and bifurcations of the adaptive exponential integrate-and-fire model. *Biol. Cybern.*, 99(4-5):319–334, 2008.
- [TD94] B TAWFIK et D.M. DURAND : Nonlinear parameter estimation by linear association : Application to a five-parameter passive neuron model. *IEEE Transactions on Biomedical Engineering*, 41:5, 1994.
- [TG08] J. H. TIEN et J. GUCKENHEIMER : Parameter estimation for bursting neural models. *Journal of Computational Neuroscience*, 24:358–373, 2008.
- [THPM06] A.L. TAYLOR, T.J. HICKEY, A.A. PRINZ et E. MARDER : Structure and visualization of high-dimensional conductance spaces. *Journal of Neurophysiology*, 96:891–905, 2006.
- [THR04] T. TATENO, A. HARSCH et H.P.C. ROBINSON : Threshold Firing Frequency-Current Relationships of Neurons in Rat Somatosensory Cortex : Type 1 and Type 2 Dynamics. *J Neurophysiol*, 92(4):2283–2294, 2004.
- [TMM00] J. TABAK, J. RICHARD MURPHEY et L.E. MOORE : Parameter estimation for bursting neural models. *Journal of Computational Neuroscience*, 9:215–236, 2000.
- [Tre03] I.C. TRELEA : The particle swarm optimization algorithm : convergence analysis and parameter selection. *Inf. Process. Lett.*, 85(6):317–325, 2003.
- [VB99] M.C. VANIER et J.M. BOWER : A comparative survey of automated parameter-search methods for compartmental neural models. *Journal of Computational Neuroscience*, 7:149–171(23), 10 September 1999.
- [VT04] J. VESTERSTROM et R. THOMSEN : A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. *In Evolutionary Computation, 2004. CEC2004.*, volume 2, pages 1980–1987, Washington, DC, USA, June 2004. IEEE Computer Society.
- [WB96] X.J. WANG et G. BUZSAKI : Gamma Oscillation by Synaptic Inhibition in a Hippocampal Interneuronal Network Model. *J. Neurosci.*, 16(20):6402–6413, 1996.
- [WBHWG99] A.R. WILLMS, D.J. BARO, R.M. HARRIS-WARRICK et J. GUCKENHEIMER : An improved parameter estimation method for Hodgkin-Huxley models. *Journal of Computational Neuroscience*, 6:145–168, 1999.
- [WC72] H. R. WILSON et J. D. COWAN : Excitatory and inhibitory interactions in localized populations of model neurons. *Biophysical Journal*, 12:1–24, Janvier 1972.
- [WCR<sup>+</sup>98] J.A. WHITE, C.C. CHOW, J. RITT, C. SOTO-TREVINO et N. KOPELL : Synchronization and oscillatory dynamics in heterogeneous, mutually inhibited neurons., 1998.
- [WM97] D. H. WOLPERT et W. G. MACREADY : No free lunch theorems for optimization. 1:67–82, 1997.

- [WSMS01] D.J. WIELAARD, M. SHELLY, D. MCCLAUGHLIN et R. SHAPLEY : How Simple Cells Are Made in a Nonlinear Network Model of the Visual Cortex. *J. Neurosci.*, 21(14):5203–5211, 2001.
- [XT08] Z. XIAO et X. TIAN : Neuronal ensemble rate coding of the simulated spike trains in the temporal lobe cortex via small-world network. *Intelligent Information Technology Applications, 2007 Workshop on*, 3:727–731, 2008.
- [You37] G. YOUNG : Note on excitation theories. *Psychometrika*, 2(2):103–106, 1937.



# Liste des publications

## Articles dans des revues avec comité de lecture

[1] H. CHEN, S. SAÏGHI, L. BUHRY, and S. RENAUD : Real-Time Simulation of Biologically Realistic Stochastic Neurons in VLSI. *IEEE Trans. on Neural Networks*, 21(9) :1511-7.

## Communications dans des conférences internationales avec comité de sélection et actes

[2] L. BUHRY, S. SAÏGHI, S. RENAUD, J. RINZEL : Cellular intrinsic properties and neural network global behaviour in the context of gamma oscillations. Soumis à *Conference on Neural Information Processing Systems (NIPS 2010)*, 2010.

[3] L. BUHRY, A. GIREMUS, E. GRIVEL, S. SAÏGHI, S. RENAUD : Automated Tuning of Analog Neuromimetic Integrated Circuits. Dans *Proceeding of the International Conference on Biomedical Circuits and Systems (BioCAS09)*. Pékin, Chine, 26-29 Novembre, (2009).

[4] L. BUHRY, A. GIREMUS, E. GRIVEL, S. SAÏGHI, S. RENAUD : New variant of differential evolution algorithm : application for neuroscientists, Dans *European Signal Processing Conference EUSIPCO 2009*. Glasgow, Écosse, 24-28 Août, (2009).

[5] L. BUHRY, S. SAÏGHI, W. BEN SALEM, S. RENAUD : Adjusting Neuron Models in Neuromimetic ICs using the Differential Evolution Algorithm. Dans *Proceeding of the International Conference on Neural Engineering (NER09)*, 28 Avril – 2 Mai, Turquie (2009).

[6] L. BUHRY, S. SAÏGHI, A. GIREMUS, E. GRIVEL, S. RENAUD : Parameter estimation of the Hodgkin-Huxley model using metaheuristics : application to neuromimetic analog integrated circuits. Dans *Proceedings of the IEEE Biomedical Circuits and Systems Conference (BIOCAS08)*, 20 – 22 Novembre, États-Unis d'Amérique (2008).

[7] S. SAÏGHI, L. BUHRY, Y. BORNAT, G. N'KAOUA, J. TOMAS, S. RENAUD : Adjusting the Neurons Models in Neuromimetic ICs using the Voltage-Clamp Technique. Dans *Proceedings of the IEEE International Symposium on Circuits And Systems 2008 (ISCAS08)*, 18 - 21 Mai, États-Unis d'Amérique (2008).

[8] A. DAOUZLI, S. SAÏGHI, L. BUHRY, Y. BORNAT, S. RENAUD : Weights Convergence and Spikes Correlation in an Adaptive Neural Network Implemented on VLSI. Dans *Proceedings of the IEEE International Conference on Bio-inspired Systems and Signal Processing (BIOSIGNALS)*, 28 - 31 Janvier, Portugal (2008).

**Communications dans des conférences nationales avec comité de sélection et actes**

[9] L. BUHRY, S. SAÏGHI, A. GIREMUS, E. GRIVEL, S. RENAUD : Estimation des paramètres du modèle d’Hodgkin–Huxley par des métaheuristiques. Dans *Actes de Congrès de la Deuxième Conférence Française de Neurosciences Computationnelles, (Neurocomp 2008)*, 8 - 11 Octobre, France (2008).

[10] L. BUHRY, S. SAÏGHI : Réglage de paramètres neuronaux par des techniques de “voltage-clamp” sur des ICs neuromimétiques. Dans *Proceedings of JNRDM 2008*, 14 - 16 Mai, France (2008).