
Rapport projet ECG

Création d'un logiciel générateur d'ECG

*En collaboration avec David Perlmutter et Eliot Crespin de chez Implicit
Encadré par Micheal Kirkpatrick, Cristian Puentes et Trungdung Le*

Réalisé par :

Sarah WU

Marie CUBAYNES

Guillaume GERONY

Sélène JEAN-MACTOUX

Hugo HAKEM

Virgile RAPEGNO

Table des matières

1	Résumé	1
1.1	Contexte et objectifs	1
1.2	Méthodologie	1
1.3	Résultats	1
1.4	Perspectives	1
2	Introduction	2
2.1	Contexte	2
2.2	Livrables attendus	2
2.3	Mise à disposition des codes	2
3	État de l'art	3
3.1	L'électrocardiogramme : normal et anormal	3
3.2	La simulation d'électrocardiogrammes aujourd'hui	4
4	Méthodologie	5
4.1	Structure du projet	5
4.2	Première Méthode : Modélisation de l'activité cardiaque par un système de 4 équations différentielles	5
4.2.1	Description de la méthode	5
4.2.2	Limites de la méthode	6
4.3	Deuxième Méthode : Modélisation cellule par cellule d'un coeur en 3 dimensions	7
4.3.0.1	Une approche tensorielle	7
4.3.0.2	Propagation du signal et obtention de l'ECG	8
4.3.0.3	Problématiques soulevées	8
4.3.0.4	Les limites	8
5	Résultats	9
5.1	Méthode des 4 équations différentielles : avis d'un expert	9
5.2	Interface graphique	10
5.3	Générateur dans le terminal	12
5.4	Méthode par modélisation informatique du coeur	12
6	Conclusion	15
6.1	Conclusion générale	15
6.2	Création de valeur	15
6.3	Perspectives	15
7	Remerciements	17
	Bibliographie	18

Chapitre 1

Résumé

1.1 Contexte et objectifs

Un électrocardiogramme (ECG) est une technique d'enregistrement graphique des courants électriques qui accompagnent les contractions du cœur, permettant de mettre en évidence des troubles du rythme ou une affection du muscle cardiaque. Leur analyse peut être périlleuse dans certains cas pour les médecins. Une intelligence artificielle permettant de détecter les anomalies dans les électrocardiogrammes pourrait rendre leur travail plus efficace et réduire les mauvais diagnostics. Cependant il existe des pathologies rares ou des combinaisons rares de pathologies pour lesquelles nous n'avons pas suffisamment de données pour entraîner une intelligence artificielle. Il est donc utile de développer un programme permettant de créer des bases d'électrocardiogrammes à la demande. De plus, un tel programme permettrait d'entraîner de futurs médecins. C'est sur ce programme que nous travaillons.

1.2 Méthodologie

Nous avons exploré deux possibilités :

- Une méthode empirique se basant sur un système de quatre équations différentielles. Une combinaison linéaire de ces quatre variables permet de modéliser un électrocardiogramme, et c'est en modifiant les différents paramètres que l'on peut simuler différentes pathologies ou combinaisons de pathologies.
- Une méthode plus explorative qui consiste à modéliser informatiquement en trois dimensions le cœur et de faire passer un influx nerveux de proche en proche dans la simulation du myocarde. Cette méthode, si probante, permettrait de modéliser directement les problèmes physiques conduisant à des pathologies (comme les blocs par exemple) et d'avoir plus facilement les électrocardiogrammes caractéristiques de ces pathologies qu'avec la première méthode.

1.3 Résultats

La première méthode a rapidement donné des résultats satisfaisants, et nous a permis d'avoir des électrocardiogrammes caractéristiques de pathologies classiques (fibrillation, tachycardie ventriculaire et auriculaire, flutter etc). Pour rendre ces enregistrements plus proches de la réalité nous avons rajouté du bruit, ce qui nous permet d'avoir de nombreux ECG pour chaque pathologies.

La deuxième méthode étant plus explorative, nous n'avons pas obtenus de résultats concrets. Dans une première approximation nous avons modélisé le cœur à l'aide de deux ellipses imbriquées et inclinées pour les ventricules et de deux boules imbriquées aux ellipses pour les oreillettes. Nous avons défini une certaine épaisseur de myocarde et des règles pour le passage de l'influx nerveux : une "cellule" s'active si une des "cellules" voisines s'active. Nous avons également implémenté le temps au bout duquel une "cellule" se désactive en fonction des régions du cœur (oreillette ou de ventricule). Pour l'instant cette modélisation est trop imprécise pour obtenir des électrocardiogrammes satisfaisants notamment car l'enveloppe du cœur est trop éloignée de la réalité et car le chemin de l'influx nerveux n'est pas tout à fait correct. Nous avons travaillé à améliorer ces points mais nous n'avons pas pu obtenir de résultats concluants.

1.4 Perspectives

Grâce à l'exploration des deux méthodes nous avons une vision plus complète du sujet, et la deuxième méthode pourrait faire l'objet d'un prochain projet.

Chapitre 2

Introduction

2.1 Contexte

Depuis quelques années, le développement de l'électronique et de l'informatique a déclenché une révolution dans de nombreux domaines. En santé, c'est la miniaturisation et l'augmentation des capacités calculatoires qui permettent l'essor de la big data et de son traitement par l'intelligence artificielle. Cependant, plusieurs problématiques entravent leur bon développement. Les données des patients sont sensibles, il est donc assez compliqué d'accéder à des bases de données denses et variées. Pire, certaines pathologies sont si rares que même sans la contrainte légale du traitement des données personnelles, trop peu d'enregistrements sont disponibles.

Or il est nécessaire d'obtenir le plus de données labélisées possibles afin d'entraîner une intelligence artificielle avec un taux d'erreurs correct. Pour répondre à cette problématique, un programme permettant d'automatiser le développement de grandes bases de données serait très pertinent.

Implicit, une entreprise spécialisée dans la cardiologie, nous a donc contactés pour créer un programme qui fournit sur demande une base de données d'électrocardiogrammes typiques de pathologies, qui leur permettrait d'entraîner une intelligence artificielle qui reconnaîtrait les pathologies et rendrait le travail des médecins plus efficace et réduirait les erreurs de diagnostic. Ce programme permettrait également d'entraîner de futurs médecins à reconnaître des pathologies à partir d'ECG.

2.2 Livrables attendus

Le rendu de notre projet est un générateur d'électrocardiogramme incluant différentes pathologies et des caractéristiques individuelles. Ce générateur doit être accessible au travers de deux logiciels aux applications différentes :

- Un générateur d'ECG réalistes afin de constituer des bases de données d'entraînement pour intelligences artificielles et ainsi aider le développement de ces technologies pour Implicit,
- Une application graphique à destination des étudiants en médecine afin de former ceux-ci à la lecture d'ECG et à l'établissement de diagnostic.

Ce générateur doit être capable de :

- Simuler un électrocardiogramme de cœur sain ou malade de façon vraisemblable,
- Inclure les pathologies suivantes : fibrillation atriale, tachycardie, flutter atrial, asystole, bradycardie, tachycardie, fibrillation ventriculaire ; et d'autres si possible,
- Tenir compte des perturbations de mesure typique (ex : bruits parasites, mouvements du patient...)

Les contraintes sur le générateur sont :

- Un générateur rédigé en python,
- Une sortie au format standard MIT-BIH (La base de données MIT-BIH est un ensemble de données accessibles au public qui fournit du matériel d'investigation standard pour la détection des pathologies cardiaques),
- Une interface graphique intuitive pour visualiser facilement les ECG générés.

2.3 Mise à disposition des codes

Trouvez ci-dessous le lien vers notre gitlab afin de télécharger et essayer notre générateur :
https://gitlab-student.centralesupelec.fr/hugo.hakem/ecg_generator

Chapitre 3

État de l'art

3.1 L'électrocardiogramme : normal et anormal

Les pathologies cardiaques sont la première cause de mortalité au monde [5], avec près de 9 millions de personnes en mourant chaque année. Les habitants des pays riches comme des pays pauvres sont largement touchés, et en France près de 4 millions de personnes sont suivies pour un problème cardiaque [1]. Il est dès lors important de pouvoir détecter au mieux ces maladies afin de poser au plus tôt un diagnostic correct et par la suite trouver le traitement adéquat.

Pour détecter une pathologie, il existe de nombreuses méthodes : échographie, IRM, radiographie, angiographie, scintigraphie. Chacune de ces méthodes permet de regarder le cœur et de chercher des signes de mal fonctionnement. Cependant, elles nécessitent des machines imposantes et chères, un technicien pour opérer la machine, un docteur pour interpréter l'image obtenue. À contrepied de ces méthodes, on trouve l'électrocardiogramme. C'est un outil d'analyse ne nécessitant que des électrodes, aujourd'hui suffisamment miniaturisé pour être porté au quotidien avec des traqueurs connectés. Un professionnel de santé peut utiliser un ECG jusqu'à 6 voies, soit 6 évolutions de différences de potentiels. Mais un ECG une voie contient déjà une grande quantité d'information.

Un ECG normal (pour un patient au repos sans aucune pathologie cardiaque) est la répétition d'un motif périodique qui commence par une onde P de faible intensité pour la dépolarisation des oreillettes. Suit le complexe QRS qui est la succession rapide d'une onde positive, puis négative puis à nouveau positive, avec une forte intensité pour la dépolarisation des ventricules. Peut s'en suivre une onde T pour la repolarisation des ventricules. Enfin il peut exister une onde U après toujours positive. [8]

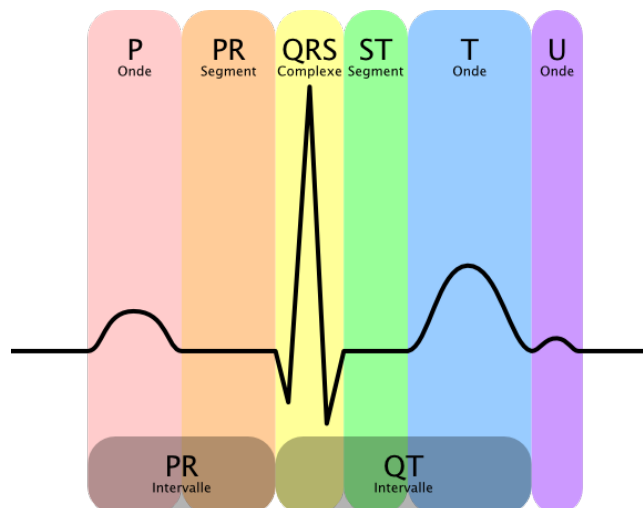


FIGURE 3.1 – Exemple d'un motif sur un ECG avec la nomenclature des ondes et des intervalles principaux, wikipédia

On trouve dans la littérature les valeurs suivantes :

- Onde P : moins de 100ms et max 2,5mV
- intervalle PR : max 200ms
- Complexe QRS : moins de 100ms

— Intervalle QT : max 430ms

On trouve ensuite le rythme cardiaque en comptant le nombre de motifs par minute, ce qui permet aussi d'établir si le patient présente des troubles du rythme cardiaque. Il existe des grilles de lecture qui, en fonction de l'écart observé à l'ECG normal, permet de déduire des maladies cardiaques probables. [4]

3.2 La simulation d'électrocardiogrammes aujourd'hui

Plusieurs méthodes sont possibles pour générer des ECG de manière artificielle :

- Une méthode utilisée est basée sur un système d'équations différentielles qui modélisent les nœuds du cœur comme des oscillateurs non linéaires. Les paramètres ont été déterminés de manière empirique par du machine learning.[7]

- On peut aussi approximer un ECG de patient sain par des fonctions simples définies par morceaux. Par exemple, une onde P peut être approchée par une fonction quadratique et le complexe QRS par une fonction triangulaire. De cette façon on réalise un ECG idéal qui sert de référence pour les ECG de personnes atteintes de pathologies.[2]

- Une façon de procéder est de résoudre un système d'équations différentielles décrivant les courants ioniques membranaires sur un modèle en 3 dimensions du cœur et du thorax. La méthode des éléments finis avec un maillage adapté permet de se rapprocher au mieux du fonctionnement réel du cœur. [6]

Chapitre 4

Méthodologie

4.1 Structure du projet

Nous avons travaillé en plusieurs phases : - Appropriation du sujet et recherches bibliographiques

- Choix des modélisations que nous pensions pertinentes pour notre projet
- Répartition des tâches entre les membres du groupe : deux personnes travaillent sur chaque modélisation, une personne travaille sur l'interface graphique, et une personne sur la conversion des données en MIT-BIH
- Mise en commun de notre travail et rédaction du rapport.
- Vérification par un cardiologue, Dr Jean, du réalisme des ECG que nous avons générés

Pour harmoniser notre travail et vérifier que nous avançons dans la bonne direction nous avons des réunions bimensuelles avec M.Perlmutter et M.Crespin.

4.2 Première Méthode : Modélisation de l'activité cardiaque par un système de 4 équations différentielles

4.2.1 Description de la méthode

Cette méthode s'appuie principalement sur l'article "Generation of ECG signals from a reaction-diffusion model spatially discretized" [7] publié dans Nature, qui fournit un système de 4 équations à 4 inconnues paramétré par une constante H de couplage entre les oscillateurs. La valeur de cette constante est déterminante pour la forme du motif obtenue.

$$\begin{aligned}\dot{x}_1 &= x_1 - x_2 - Cx_1x_2 - x_1x_2^2, \\ \dot{x}_2 &= Hx_1 - 3x_2 + Cx_1x_2 + x_1x_2^2 + \beta(x_4 - x_2), \\ \dot{x}_3 &= x_3 - x_4 - Cx_3x_4 - x_3x_4^2, \\ \dot{x}_4 &= Hx_3 - 3x_4 + Cx_3x_4 + x_3x_4^2 + 2\beta(x_2 - x_4).\end{aligned}$$

L'ECG est obtenu comme combinaison linéaire des 4 inconnues du système : $ECG(t) = \alpha_1x_1 + \alpha_2x_2 + \alpha_3x_3 + \alpha_4x_4$.

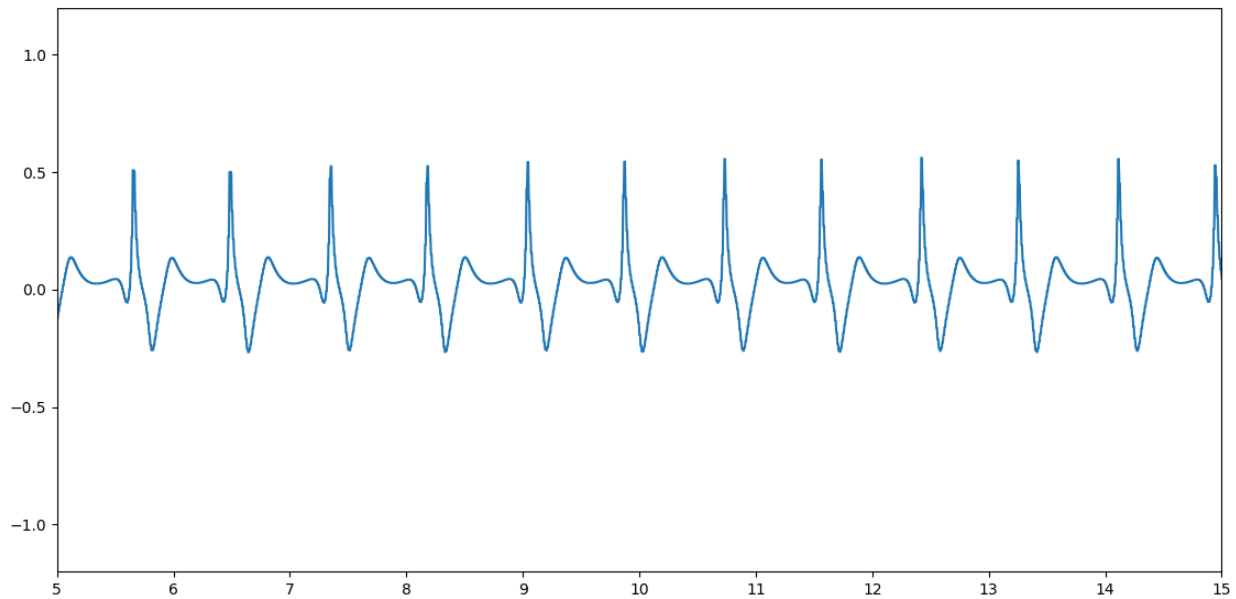


FIGURE 4.1 – Exemple d'un ECG d'une personne saine obtenu par la résolution du système

Le signal brut obtenu ne correspondant pas à ceux pouvant être observés dans la réalité, la deuxième étape a consisté à traiter le signal pour lui rajouter les 4 principales sources de bruit : [9] :

- le bruit de type Baseline Wander (BW) qui déforme la ligne de tendance du signal, modélisé par une somme de cosinus
- le bruit blanc gaussien (WGN) modélisé par la réalisation d'une variable de loi normale de moyenne nulle à chaque instant
- le bruit de type Muscle Artifact (MA) qui correspond à une amplification du signal sur de courts instants à cause d'un mouvement du patient modélisé comme le bruit blanc sur des intervalles tirés aléatoirement
- le bruit de type Power Line Interference (PLI) causé par des interférences du réseau électrique et créant des pics à haute fréquence sur le signal, modélisé par un cosinus de fréquence 50 Hz.

Pour simuler des ECG différents les uns des autres, la fréquence cardiaque est tirée aléatoirement à chaque génération d'ECG, ainsi que les conditions initiales des inconnues, et les paramètres des différents bruits. De plus le paramètre H est également légèrement perturbé à chaque fois, ce qui change le motif brut de l'ECG.

Enfin nous avons travaillé à la programmation d'une interface graphique permettant de choisir le type de pathologie souhaitée (ou non), la probabilité de présence de certains bruits, et l'activation de la détection automatique des ondes R. Cette interface a été programmée grâce à la librairie TKinter sous Python [3].

4.2.2 Limites de la méthode

Si la méthode possède l'avantage d'être facilement implémentable et solvable, sa limite principale réside dans le fait qu'on ne comprend pas bien comment la variation des paramètres (notamment H) affecte le motif de l'ECG. Ainsi, nous sommes restés pour l'instant dépendant de l'article en utilisant des valeurs pour H proches de celles recommandées, et en incluant seulement les pathologies pour lesquelles les paramètres étaient fournis. Ces paramètres ont été obtenus par les chercheurs en minimisant l'écart entre des ECG réels de personnes malades et ceux simulés par la résolution du système. Cette piste peut être creusée de notre côté pour tenter de simuler d'autres pathologies.

4.3 Deuxième Méthode : Modélisation cellule par cellule d'un coeur en 3 dimensions

Cette seconde méthode est le fruit d'une discussion avec M. Perlmutter et M. Crespin . L'approche qu'ils nous ont proposée est de partir d'une modélisation d'un groupe de cellules cardiaques pour ensuite la généraliser à l'enveloppe d'un coeur. Cette démarche permet en effet de travailler plus dans le détail en contrôlant la façon dont un groupe de cellules réagit à l'influx nerveux et la façon dont son dysfonctionnement se répercute sur le coeur tout en entier. L'idée qui réside derrière cette approche est en définitive très intuitive :

Connaissant la manière dont une maladie cardiaque affecte une région du coeur ou le chemin suivi par l'influx nerveux, il devient possible d'agir avec précision et de modéliser plus facilement les maladies cardiaques inaccessibles par la première méthode, comme les blocs par exemple.

De plus, il devient possible d'influer sur des paramètres comme la taille du coeur ou sa morphologie.

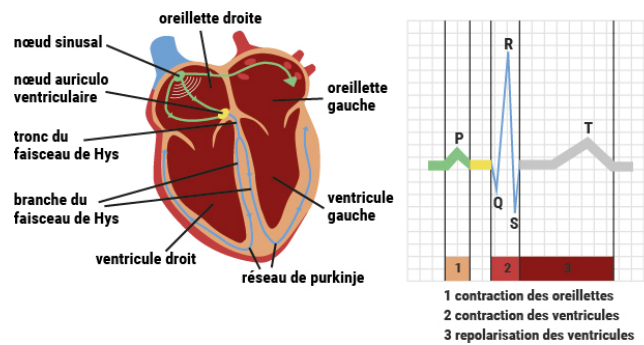


FIGURE 4.2 – Schéma du coeur et des éléments clés de son activité électrique

4.3.0.1 Une approche tensorielle

Afin de modéliser le coeur cellule par cellule, nous avons choisi d'utiliser un tenseur à 4 dimensions. 3 dimensions pour discrétiser l'espace, et 1 dimension pour indiquer les propriétés de la cellule. Ainsi :

1. $A[x][y][z][0]$ = la case appartient oui (1) ou non (0) au coeur (la matrice étant par essence un cube, pour pouvoir modéliser un coeur, il faut "dessiner" ce coeur dans le cube)
2. $A[x][y][z][1]$ = la cellule est activée (1) ou non (0) par l'influx nerveux.
3. $A[x][y][z][2]$ = caractérise la cellule (si celle ci met longtemps à s'exciter, elle s'activera avec un retard, si elle s'excite avec une plus grande intensité que les autres cellules, on associera un coefficient à l'excitation.)

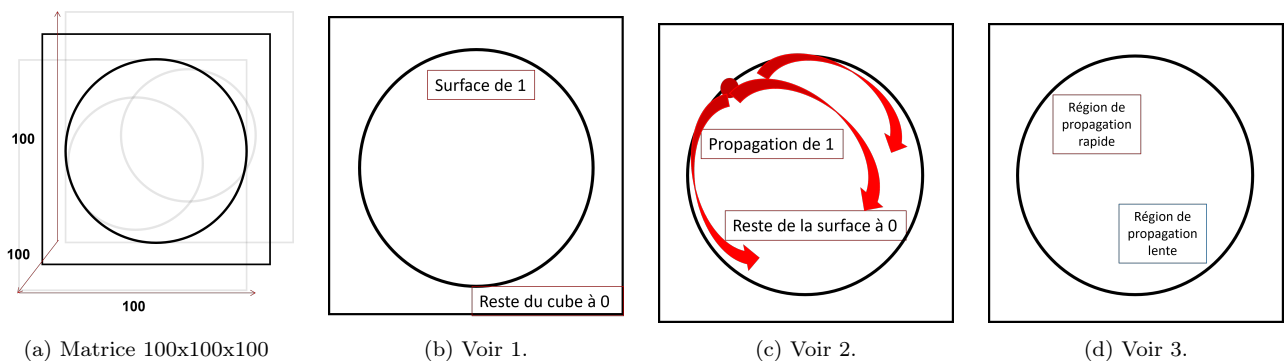


FIGURE 4.3 – Schéma décrivant la modélisation tensorielle utilisée

4.3.0.2 Propagation du signal et obtention de l'ECG

Pour modéliser la propagation du signal électrique nous avons choisi de reprendre le principe du jeu de la vie. C'est à dire qu'une cellule s'active lorsqu'une cellule à côté d'elle est activée. Nous supposons finalement que l'ECG est la somme des amplitudes d'activation de chaque cellule. De la même façon, l'ECG est une moyenne des différents électrogrammes (EGM) au plus proche du coeur.

4.3.0.3 Problématiques soulevées

1. Comment modéliser l'enveloppe du coeur ?

Nous avons commencé simplement par une sphère, puis par deux sphères et deux ellipsoïdes droites imbriquées afin de décrire les oreillettes et les ventricules, puis nous avons incliné ces ellipsoïdes afin de s'approcher d'avantage de la forme d'un coeur.

2. Comment modéliser le réseau de Purkinje ?

L'influx nerveux se propage dans l'épaisseur de myocarde que nous avons modélisé. Nous devons nous assurer que l'influx nerveux passe bien au bon endroit, c'est à dire qu'il ne se propage pas sur l'extérieur entre les oreillettes et les ventricules. Pour cela nous avons "supprimé" des cellules en disant qu'elles n'appartenaient pas au coeur pour stopper la propagation au bon endroit.

3. Quels temps d'activation et coefficient utiliser pour chaque cellule ? Nous avons utilisé les données de l'article [6] pour attribuer à chaque type de cellules un certain temps d'activation et un coefficient en fonction de l'intensité (nous avons gardé tous ces coefficients égaux à 1).

4. Comment retranscrire un nombre de bit en potentiel ?

Finalement se pose également la question de comment visualiser notre modélisation matricielle du coeur. Ploter un tenseur 3D n'est pas évident en python, tandis que cela est bien plus simple pour des matrices 2D. C'est pourquoi nous avons cherché à écrire un code qui permet une visualisation d'une coupe du coeur. Nous avons par ailleurs ajouté une fonctionnalité permettant de visualiser la propagation de l'influx nerveux. De cette façon modifier notre approche et redimensionner le coeur matricielle est plus simple.

4.3.0.4 Les limites

Modéliser un coeur à l'aide d'une matrice n'est en définitive pas une chose facile. Nous avons travaillé de façon explorative en testant, et en ayant une idée des résultats que nous sommes censés obtenir après chaque modification. Les résultats ne sont cependant pas ceux escomptés.

Par ailleurs, il faut énormément de données pour modéliser avec précision un coeur. Notre modélisation avec des sphères et des ellipses nous a conduit à utiliser un tenseur de dimension $100*100*100*3$ ce qui correspond à 3 millions de données et ce qui commence à être difficilement acceptable par Python. Nos codes mettent en effet entre 3 et 5 minutes à s'exécuter ce qui pose des problèmes d'efficacité, et par la suite de résolution.

Une façon d'y pallier serait d'optimiser nos codes avec les outils de la bibliothèque NumPy fournit par Python. Une autre façon serait de s'affranchir de toutes les dimensions et de travailler en 2D. En effet, notre choix d'évolution pourrait aussi se faire à l'aide d'un graphe où chaque lien relie une cellule à une cellule voisine. Il faut alors réfléchir à la transformation de notre modèle 3D en sa projection 2D, soit la conversion de notre tenseur en un type de données plus approprié. Un gain calculatoire conséquent est à chercher de ce côté.

Nous avons travaillé sur la diversité des types de cellules. Ceci pose une nouvelle fois un problème d'optimisation car là où on faisait passer l'influx nerveux facilement d'une cellule à une autre, il faut désormais prendre en compte son délai de réponse ou de désactivation et donc retenir que l'information est arrivée jusqu'à la cellule et qu'elle doit donc réagir avec un temps de latence.

Ceci complexifie le code et donc allonge son temps d'exécution en l'état de nos choix d'implémentation.

Chapitre 5

Résultats

5.1 Méthode des 4 équations différentielles : avis d'un expert

Pour tester les résultats obtenus pour la méthode des 4 équations différentielles nous avons montré les ECG obtenus à Frédéric Jean, un cardiologue du CHU de Clermont-Ferrand (dont voici quelques exemples ci-dessous).

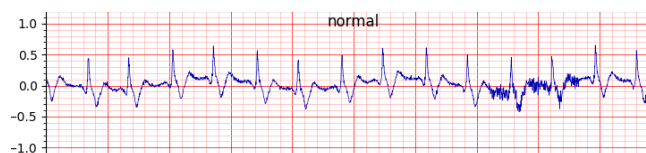


FIGURE 5.1 – ECG d'une personne saine sans bruit de type PLI



FIGURE 5.2 – ECG d'une personne saine avec bruit de type PLI



FIGURE 5.3 – ECG d'une personne atteinte de flutter atrial sans bruit de type PLI

Selon le docteur Jean le bruit qu'on a ajouté est globalement cohérent, il est parfois trop accentué quand l'amplitude des signaux est trop faible ce qui rend compliqué la lecture de l'électrocardiogramme. Pour remédier à ce problème d'amplitude nous pouvons rajouter un gain qui augmente l'amplitude du signal. De plus, de manière générale on voit trop peu l'onde P qui peut rendre le diagnostic rythmique compliqué.

Pour ce qui est des caractéristiques des ECG : les ECG normaux, de tachycardie sinusale, de bradycardie, d'asystolie et de flutter atrial sont très cohérents, ils correspondent très bien à ce qu'on peut voir chez des patients et sont utilisables en l'état.

Pour la tachycardie ventriculaire et le flutter ventriculaire les signaux ont une amplitude trop faible et se ressemblent trop.

La fibrillation ventriculaire ressemble trop à une fibrillation atriale parce ce qu'on identifie clairement les QRS, donc le signal est trop organisé et pas anarchique.

Finalement pour la fibrillation atriale, certaines parties ne sont pas cohérentes (on a l'impression que c'est du bruit) et il faudrait que les QRS soient irréguliers.

A part pour la tachycardie et le flutter ventriculaire le docteur Jean a pu reconnaître les pathologies. Ainsi le logiciel est adapté à l'enseignement des étudiants en médecine. Pour ce qui est de l'entraînement de l'intelligence

artificielle, certaines modifications sont nécessaires mais elles sont peu nombreuses.

5.2 Interface graphique

Nous avons programmé en utilisant tkinter une interface graphique permettant de générer les ECG souhaités par l'utilisateur.

L'utilisateur peut alors rentrer la probabilité de présence des bruits PLI et MA sur l'ECG représentant la pathologie qu'il souhaite.

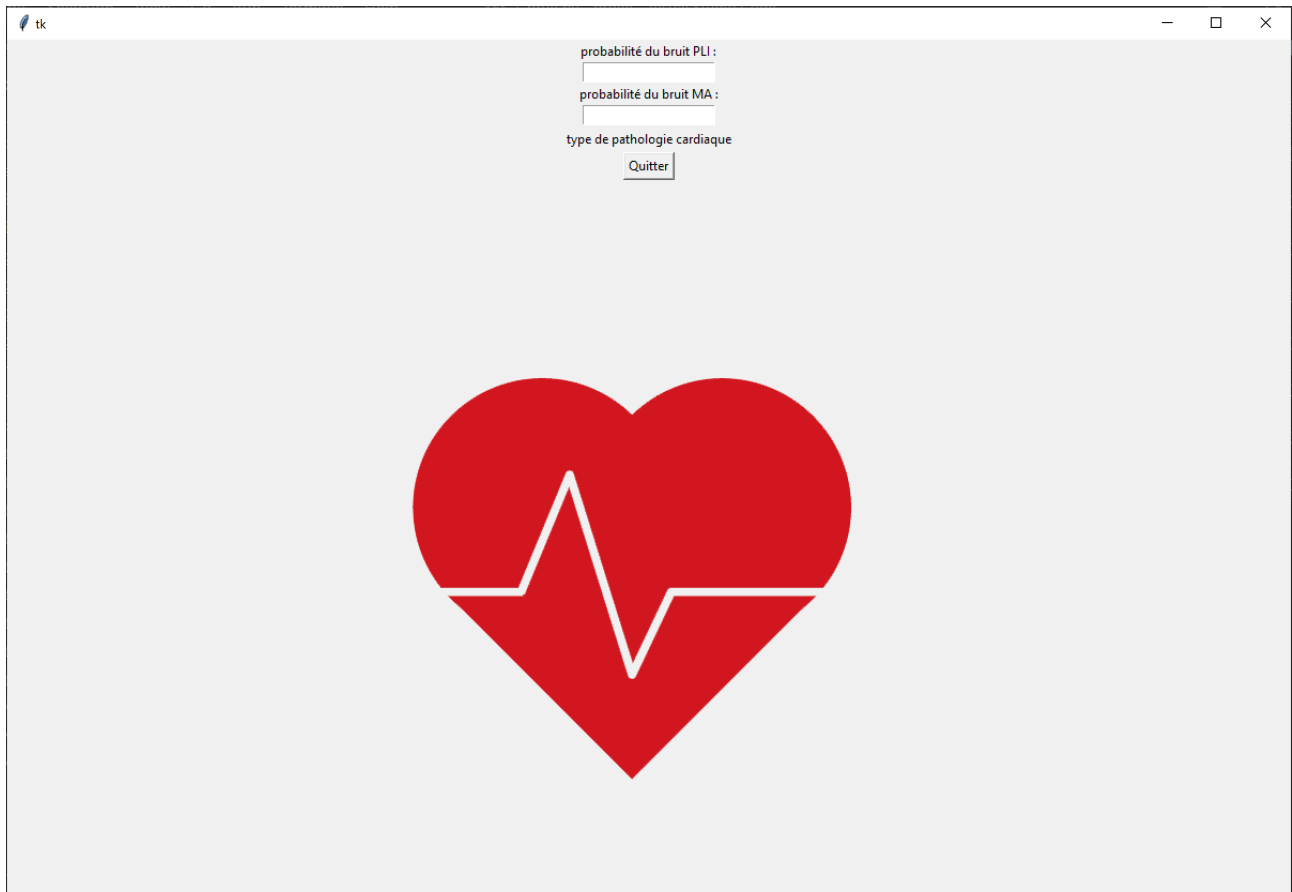


FIGURE 5.4 – Fenêtre d'accueil de l'interface graphique

probabilité du bruit PLI :

probabilité du bruit MA :

type de pathologie cardiaque

normal
vtachy
atrflutter
sintachy
vfib
vflutter
asystolie
brady
atrfib

FIGURE 5.5 – Menu déroulant des différentes pathologies

Quelques exemples d'ECG générés par un utilisateur selon les données rentrées :

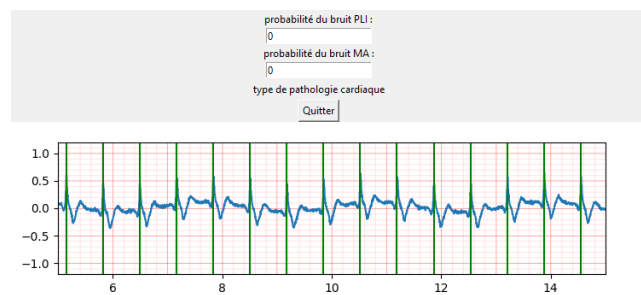


FIGURE 5.6 – ECG d'une personne saine sans aucun bruit

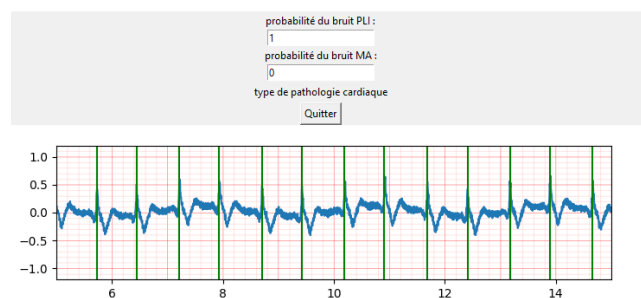


FIGURE 5.7 – ECG d'une personne atteinte de flutter atrial avec du bruit PLI seulement

Dans une prochaine version, l'utilisateur pourra choisir d'afficher ou non des repères indiquant la présence des ondes R afin de l'aider dans son analyse de l'ECG :

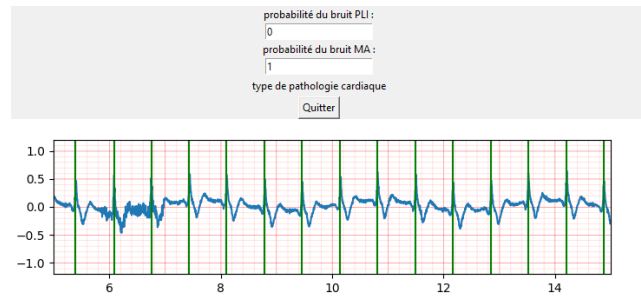


FIGURE 5.8 – ECG d’une personne atteinte de tachycardie ventriculaire avec du bruit MA

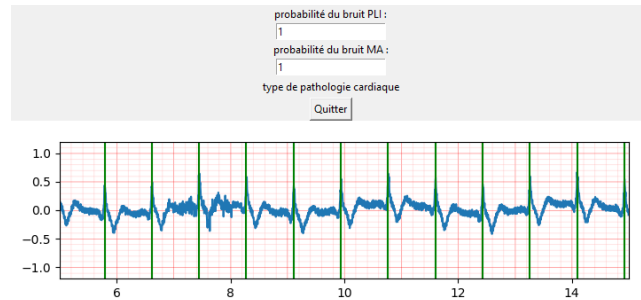


FIGURE 5.9 – ECG d’une personne atteinte de bradycardie avec les deux types de bruit

probabilité du bruit PLI :

 probabilité du bruit MA :

☒ afficher les ondes R ?
 type de pathologie cardiaque

5.3 Générateur dans le terminal

Afin de profiter au mieux de notre générateur, il était nécessaire de développer un outil pour sortir en quantité des ECG. Ainsi il est possible en utilisant `export.py` de générer des ECG depuis le terminal en définissant par avance les caractéristiques désirées.

L’utilisation s’effectue en passant dans l’ordre comme argument :le nom de la maladie, le nombre d’ECG désiré et la localisation (on peut compléter avec `-PLI` et `-MA` pour indiquer la probabilité respective des bruit de Power line Wander et Muscle Artifact).

L’export est alors au format MIT-BIH, le standard utilisé par Impicity pour leur algorithme de machine learning.

5.4 Méthode par modélisation informatique du coeur

Nous avons modélisé le passage de l’influx nerveux dans le myocarde

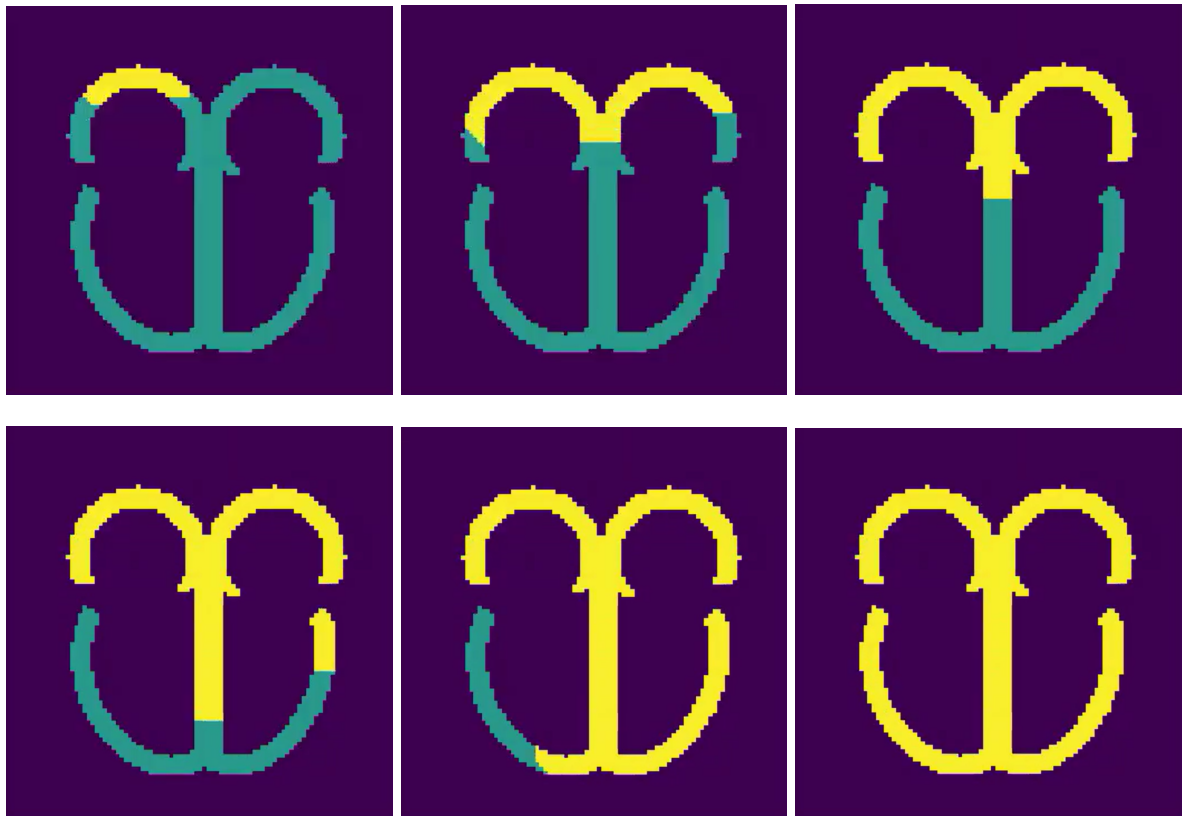


FIGURE 5.11 – Propagation d'un influx nerveux dans le myocarde

Pour ce qui est de l'anatomie du coeur :

- Nous avons "coupé" le coeur entre les oreillettes et les ventricules pour que l'influx nerveux ne passe pas par cet endroit là mais par la cloison interventriculaire. Cependant l'influx passe tout de même au mauvais endroit entre l'oreillette gauche et le ventricule gauche.
- L'anatomie du coeur n'est pas respectée, il trop symétrique par exemple mais les asymétries sont compliquées à reproduire fidèlement avec notre méthode de modélisation.

Pour ce qui est de l'influx nerveux :

Nous n'avons modélisé que la propagation dans un cas où les "cellules" restent activées (ne se désactivent pas) cependant nous avons bien pris en compte le fait qu'elles se désactivent pour l'électrocardiogramme.

Pour ce qui est de l'électrocardiogramme, nous avons tout d'abord obtenu le résultat ci-dessous pour un tau (nombre de simulations avant la désactivation de chaque cellule) de 4 et une modélisation primaire du coeur.

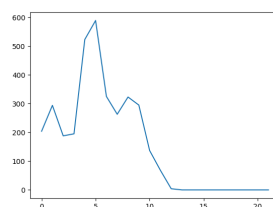


FIGURE 5.12 – Résultat pour tau=4, le 21 mars

On peut voir que la base n'est pas à 0, mais on peut voir 3 semblants de signaux ressemblants aux ondes P, QRS et T. Cela semble être un coup de chance. En effet lorsque l'on a cherché à améliorer la simulation en améliorant l'anatomie du coeur (tenté de retranscrire l'asymétrie, d'améliorer les proportions etc) et de prendre en compte les temps d'activation différents des cellules selon que ce soit des cellules de ventricule ou d'oreillette (les cellules de ventricules restent activées 1,5 fois plus longtemps que les cellules d'oreillette), nous avons obtenu le résultat suivant :

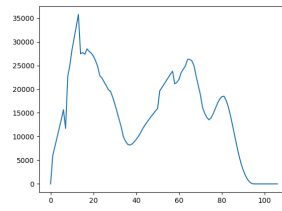


FIGURE 5.13 – Résultat pour $\tau=4$, le 11 avril

Quel que soit le τ que l'on choisit nous n'obtenons rien de proche de la réalité.

Chapitre 6

Conclusion

6.1 Conclusion générale

Pour conclure, deux approches différentes ont été développées pour répondre à la problématique posée par Implicit : Elaborer un algorithme capable de générer un grands nombres d'ECG **différents** et pouvant décrire **plusieurs maladies** pour pouvoir entraîner une Intelligence Artificielle à les reconnaître.

La première méthode utilisant un jeu de quatre équations différentielles s'est donc montrée satisfaisante, modélisant une dizaine de maladies différentes. Toutefois il est difficile avec celle-ci de modéliser de nouvelles maladies, puisqu'obtenir les bons paramètres les décrivant est difficile ces derniers n'ayant pas une réelle interprétation physique.

La seconde méthode en revanche, bien que pouvant répondre à cet enjeux de modéliser de nouvelle maladies inaccessibles par la première méthode, n'a pas été concluante. Cette méthode explorative, ne s'appuyant sur aucune publication, cherchait à décrire le coeur matriciellement. Mais alors que nous progressions sur sa modélisation en affinant la géométrie du coeur et la propagation du signal nerveux, les résultats étaient de moins en moins probants. Le fait que cette méthode dépende de nombreux paramètres (le temps de mise en activation ou d'activation de chaque cellule, ou l'intensité de chaque activation) et que nous ne possédions aucune base pour pouvoir les déterminer n'a pas facilité la tâche. Nous restons néanmoins convaincus que cette approche mérite d'être creusée, ne serait-ce que par le niveau de détail qu'elle permettrait d'obtenir, en pouvant agir avec précision sur chaque domaine du coeur ou directement sur le trajet suivi par l'influx nerveux.

6.2 Création de valeur

En sélectionnant la première méthode pour notre algorithme final, notre générateur d'ECG répond ainsi à la presque-entière de la demande d'Implicit. Nous avons en effet mis au point une méthode de génération d'un grand nombre d'ECG réalistes tenant compte d'une certaine maladie sélectionnée en entrée.

Par ailleurs, une interface graphique a également été développée, ce qui d'un point de vue pédagogique, pourrait être utile pour nos futurs médecins, ces derniers pouvant s'entraîner à reconnaître les pathologies.

En effet, notre générateur fut mis à l'épreuve par Frédéric Jean, rythmologue au CHU de Clermont-Ferrand. Bien que des modifications soient nécessaires pour correspondre davantage à la réalité, celui-ci attesta qu'il s'agissait d'une bonne première base et que certaines maladies donnent des résultats tout à fait satisfaisants.

Cependant, la liste des maladies modélisées par notre algorithme est incomplète en comparaison avec celle désirée initialement par Implicit.

6.3 Perspectives

À la suite de cette première étape dans le développement de notre projet, il convient de considérer les pistes suivantes pour perfectionner nos résultats :

Pour la première méthode, une analyse plus poussée du système d'équations est nécessaire afin de trouver une procédure expérimentale pour trouver les bons coefficients de génération pour de nouvelles pathologies. On peut envisager aussi l'utilisation d'une intelligence artificielle pour trouver ces paramètres, mais cela semble difficile puisque notre générateur doit justement pallier le manque de données pour entraîner les intelligences artificielles.

Pour la seconde méthode, il serait intéressant d'affiner la modélisation tensorielle du coeur. Cela peut passer par une réécriture de notre générateur en prenant en compte un nouveau moyen du stockage des informations.

La matrice étant globalement creuse, une mise à jour de l'état du coeur prend plus de temps que nécessaire car les informations d'intérêts ne représentent qu'une infime portion de la matrice. Il s'agirait donc de faire la transformation du 3D au 2D avec l'utilisation de graphe pour modéliser de façon plus optimisée notre coeur.

Chapitre 7

Remerciements

Merci à David Perlmutter et Eliot Crespín d'Implicit pour leur suivi tout au long du projet et leurs précieux conseils et retours chaque semaine.

Merci au docteur Jean d'avoir apporté son expertise sur les résultats obtenus avec la méthode des 4 équations différentielles.

Merci à Michael Kirkpatrick et à Cristian Puentes pour l'encadrement du projet.

Merci à Trung Dung Le d'avoir accepté d'être notre référent pédagogique, d'avoir pris le temps de nous rencontrer et pour la relécture de notre rapport.

Bibliographie

- [1] Ministère des Solidarités et de la Santé. *Maladies cardiovasculaires*.
- [2] Radana Kahankova Jan Kubicek, Marek Penhaker. Design of a synthetic ecg signal based on the fourier series.
- [3] Fredrik Lundh. An introduction to tkinter. *URL : www.pythonware.com/library/tkinter/introduction/index.htm*, 1999.
- [4] Manuel MSD. *Interprétation des ECG anormaux*.
- [5] World Health Organization. *Global Health Estimates : Life expectancy and leading causes of death and disability*.
- [6] Charles Pierre. Modélisation et simulation de l'activité électrique du coeur dans le thorax, analyse numérique et méthodes de volumes finis., 2005.
- [7] Jiménez-Ramírez O. Vázquez-Medina R. et al. Quiroz-Juárez, M.A. Generation of ecg signals from a reaction-diffusion model spatially discretized., 2019.
- [8] Christophe Scavée. Rappels des principes fondamentaux en électrocardiographie.
- [9] M Sugadev, Malladi Kaushik, V Vijaykumar, and T Ravi. Survey on various noise sources in ecg signal and its filtering methods. In *2021 10th International Conference on Internet of Everything, Microwave Engineering, Communication and Networks (IEMECON)*, pages 01–06, 2021.