

Implémentation d'une méthode d'identification de l'occupation électronique d'une boîte quantique grâce à des techniques d'apprentissage profond

par

Marc-Antoine Genest

Mémoire présenté au département de physique
en vue de l'obtention du grade de maître ès sciences (M.Sc.)

FACULTÉ des SCIENCES
UNIVERSITÉ de SHERBROOKE

Sherbrooke, Québec, Canada, 13 janvier 2020

Le 13 janvier 2020

le jury a accepté le mémoire de Monsieur Marc-Antoine Genest dans sa version finale.

Membres du jury

Professeur Michel Pioro-Ladrière
Directeur de recherche
Département de physique

Professeur Ion Garate
Membre interne
Département de physique

Professeure Eva Dupont-Ferrier
Présidente rapporteuse
Département de physique

À mes parents et amis

Sommaire

Ce mémoire présente une nouvelle méthode d'identification des lignes de transition sur un diagramme de stabilité, étape cruciale pour l'automatisation d'une boîte quantique dans le régime à un électron. Le programme implémenté est constitué d'un générateur de données expérimentales, et d'un algorithme d'apprentissage s'entraînant à l'aide de ces données. Ce dernier se veut général et indépendant du dispositif mesuré, étant basé uniquement sur la reconnaissance de transitions électroniques sur un diagramme de stabilité préalablement transformé en base deux (0 et 1).

Pour ce faire, un simulateur par éléments finis ainsi que diverses fonctions de traitement d'images ont été implémentés afin de créer rapidement la base de données d'entraînement d'un algorithme d'apprentissage identifiant les lignes de transition sur de véritables diagrammes de stabilité. Le rôle de ce programme se veut un module robuste d'identification, pixel à pixel, de diagrammes de stabilité, bénéficiant des avantages de l'apprentissage automatique. Un tel module permet, en plus de générer facilement des données d'entraînement personnalisées et générées de toutes pièces par ordinateur, d'identifier sans perdre de temps supplémentaire l'entièreté des pixels d'un diagramme de stabilité.

Le développement de ces outils constitue d'abord une nouvelle expertise dans le groupe du professeur Pioro-Ladrière, lesquels pouvant être directement appliqués aux travaux en cours. De plus, il s'agit de techniques encore jamais utilisées pour la création de données d'entraînement d'algorithme d'apprentissage, ainsi que de toutes nouvelles applications aux réseaux neuronaux, élargissant ainsi le champ d'expertise de ce récent domaine.

Ce travail est une avancée importante pour l'automatisation de l'initialisation d'un qubit de spin, rendant robuste et complète l'analyse du signal provenant d'une boîte quantique. Il permettra aussi de faciliter la généralisation à plusieurs boîtes quantiques et d'être indépendant du dispositif utilisé. Les expérimentateurs pourront donc sauver de nombreuses heures de tests visant à déterminer si un dispositif est fonctionnel, et à en trouver le régime d'opération, le cas échéant.

Remerciements

Dans un premier temps, j'aimerais remercier mon directeur de recherche, Michel Pioro-Ladrière, pour l'opportunité de maîtrise qu'il m'a offert. Michel est un motivateur et un visionnaire, avec qui on gagne beaucoup à travailler. Une maîtrise est courte pour pouvoir profiter d'une telle expertise, mais tes conseils et ta passion auront été d'une grande inspiration.

Je profiterais de l'occasion pour remercier le professeur Félix Camirand Lemyre pour sa collaboration et l'apport de ses idées novatrices en terme d'intelligence artificielle. Merci également à son étudiant à la maîtrise, Olivier Germain, avec qui prendre un simple café peut faire grandement avancer un projet.

J'aimerais ensuite remercier les membres du groupe de recherche. Particulièrement Julien Camirand Lemyre et Maxime Lapointe-Major, qui ont pris tout le temps nécessaire pour répondre à mes questions et m'aider à régler de nombreux problèmes et embûches.

Par la suite, je voudrais remercier spécialement Charles Authier, étudiant à la maîtrise en informatique, sans qui la date inscrite sur la page de présentation et les résultats de ce mémoire seraient probablement différents. J'ajouterais un merci à ma famille et amis pour le support que vous m'avez apporté, particulièrement ma mère qui a lu et commenté les quelques 70 pages de grande littérature qui suivront.

Finalement, je remercierais l'*Institut quantique*, d'abord pour l'appui financier et ainsi avoir permis le développement de ce projet, mais également pour avoir créé un milieu vivant dans lequel le réaliser.

Table des matières

Sommaire	ii
Introduction	1
1 Mesure et initialisation d'une boîte quantique	4
1.1 Définition de la boîte quantique	4
1.1.1 Modèle simplifié	4
1.1.2 Approche expérimentale	6
1.2 Initialisation de la boîte quantique dans le régime du qubit	7
1.2.1 Transistor mono-électronique et détection de charges	8
1.2.2 Mesure d'un diagramme de stabilité	9
2 Cahier des charges	12
2.1 Problématique	12
2.2 Travaux antérieurs	13
2.2.1 Recherche de formes connues	13
2.2.2 Recherche de lignes	14
2.2.3 Techniques d'apprentissage	14
2.2.4 Combinaison de reconnaissance de formes et d'algorithme d'appren- tissage	15
2.3 Méthode proposée	16
2.3.1 Objectifs et restrictions	17
2.3.2 Justification et philosophie de l'algorithme	18
3 Implémentation du simulateur	20
3.1 Profil de potentiel électrostatique	21
3.2 Profil de densité électronique	22
3.3 Distribution aléatoire de charges	23
3.4 Calcul de l'occupation électronique	25

3.5	Stratégies d'accélération pour la simulation d'un diagramme de stabilité . . .	26
3.5.1	Critères de convergence de la densité électronique	26
3.5.2	Pré-calcul de l'intégrale de Fermi-Dirac	27
3.5.3	Pré-calcul des poids d'interactions électroniques	28
3.5.4	Calcul partiel du diagramme de stabilité	28
3.6	Liste des paramètres d'entrées du modèle physique	30
4	Augmentation de données par techniques de traitement d'images	33
4.1	Passage de l'occupation électronique à la mesure expérimentale	34
4.2	Ajouts de défauts physiques dans le dispositif	36
4.2.1	Un défaut couplé au détecteur de charges, mais non à la boîte quantique	36
4.2.2	Un défaut couplé à la boîte quantique, mais non au détecteur de charges	38
4.2.3	Un défaut couplé au détecteur de charges et à la boîte quantique . . .	40
4.3	Effets du taux tunnel <i>réservoir - boîte quantique</i>	44
4.3.1	Transmission de la barrière tunnel	44
4.3.2	Taux tunnel <i>vs</i> fréquence des mesures	45
4.4	Effet de couplage dans un diagramme à double boîtes quantiques	50
4.5	Effet du pré-traitement des diagrammes de stabilité expérimentaux	52
4.6	Ajout de pixels <i>faux-positifs</i>	54
4.6.1	Par distribution uniforme de bruit	55
4.6.2	Par apprentissage profond	55
4.7	Combinaison d'artefacts et liste des paramètres	57
5	Technique d'apprentissage profond de segmentation d'images pour l'identification de l'occupation électronique	61
5.1	La vision par ordinateur aujourd'hui	62
5.2	Principe de fonctionnement des algorithmes de segmentation	63
5.2.1	Données d'entraînement	63
5.2.2	Algorithme et entraînement	64
5.3	Application sur les diagrammes des stabilité	65
5.3.1	Génération de la vérité terrain	65
5.3.2	Création de la base de données d'entraînement	66
5.3.3	Paramètres de tests et résultats	68
6	Perspectives	70
6.1	Optimisation de l'algorithme	71
6.2	Multidimensionnalité de l'algorithme	71
6.2.1	Expansion à plusieurs boîtes quantiques	72

<i>Table des matières</i>	vii
6.2.2 Expansion à plusieurs grilles de contrôle	72
6.3 Amélioration en temps réel de la performance de l'algorithme	73
6.3.1 Modification de la base de données d'entraînement	73
6.3.2 Ajout de données expérimentales à la base de données d'entraînement	74
Conclusion	74
Bibliographie	76

Liste des tableaux

3.1	Liste des paramètres d'entrée relatifs au dispositif pour le calcul d'occupations électroniques et de diagrammes de stabilité.	30
3.2	Liste des paramètres d'entrée relatifs aux grilles pour le calcul d'occupations électroniques et de diagrammes de stabilité.	31
4.1	Liste des distributions de paramètres d'entrée des différentes fonctions de traitement d'images utilisées pour la création de la base de données d'entraînement de l'algorithme de segmentation de diagrammes de stabilité.	60

Table des figures

1.1	Circuit électrique décrivant schématiquement une boîte quantique. Ici, la source est séparée de l'île par une jonction tunnel (de capacité C_s et résistance tunnel R_s), q_1 est la charge d'une plaque du condensateur de capacité C , q_2 est la charge d'une plaque du condensateur de capacité C_s , et V_g est le potentiel appliqué à la source.	5
1.2	Représentation graphique des rapports d'énergies E_{el} et E_c pour $N = -1, 0$ et 1 . À noter que le nombre d'électrons à l'intérieur de l'île serait modifié à chaque endroit où il y a une ligne pointillée rouge. Ces lignes arrivent d'ailleurs à chaque valeur demi-entière de q/e , expliquant le terme $1/2$ de l'équation 1.4.	6
1.3	Schéma d'un dispositif à boîte quantique MOS.	6
1.4	Schéma d'un dispositif à boîte quantique MOS.	8
1.5	Schéma de la courbe IV d'un transistor mono-électronique faisant la mesure d'une boîte quantique pour différentes occupations[1]. La courbe rouge peut s'obtenir grâce à l'équation $I_{TME} = \sin(V_g)$, la bleu grâce à $I_{TME} = \sin(V_g + \phi)$ ou ϕ est un retour de phase, et la jaune avec $I_{TME} = \sin(V_g + \Phi)$ où $\Phi = \phi \times \text{une fonction escalier}$	9
1.6	Schéma de la courbe IV d'un transistor mono-électronique faisant la mesure d'une boîte quantique ainsi qu'un exemple de diagramme de stabilité lui étant associé (simulé). Les lignes rouge pointillées lient le retour de phase dans le courant du TME à la transition électronique associée.	10
3.1	Schéma d'une grille de potentiel. La direction x dans cette figure, et utilisé dans le reste des équations, sort de la page (\odot)	21
3.2	Exemple d'une matrice de $K(x, x')$ calculée à l'avance selon l'équation 3.4, avec 201 éléments finis, $K_0 = 5e - 3$ eV, et $\sigma = 5e - 9$ m	28

4.1	Exemple d'une occupation électronique transformée en mesure binaire. À noter que les lignes sur l'image de droite sont supposées continues, et que les brisures ici présentes sont des artefacts visuels dues à l'extraction de la figure.	35
4.2	Exemple d'un diagramme de stabilité expérimental où une charge parasite s'est couplée au détecteur de charge pendant la mesure.	38
4.3	Exemple d'un diagramme de stabilité expérimental où une charge parasite s'est couplée à la boîte quantique pendant la mesure.	40
4.4	Exemple du résultat de la fonction de traitement d'images représentant un défaut couplé à la boîte quantique mais non au détecteur de charge. Le triangle en pointillé rouge sur le diagramme de gauche représente la translation à faire, et le résultat est affiché à droite.	40
4.5	Exemple d'un «X» se transformant en deux hyperboles.	41
4.6	Utilisation de la génération d'hyperbole Steiner pour créer l'anticroisement.	41
4.7	Exemple de résultats de la fonction de traitement d'images représentant un défaut couplé à la boîte quantique et au détecteur de charges.	41
4.8	Exemple de l'élargissement des lignes d'un diagramme de stabilité. À noter que les lignes sur l'image de gauche sont supposées continues, et que les brisures ici présentes sont des artefacts visuels dues à l'extraction de la figure.	45
4.9	Exemple d'un diagramme de stabilité expérimental (haut) où la mesure était trop rapide, et d'une simulation (bas) modifiée par l'algorithme 11. Les lignes rouges délimitent l'impact de la mesure trop rapide.	50
4.10	Organigramme de l'algorithme ajoutant les effets de double boîte quantique au diagramme de stabilité.	51
4.11	Étapes permettant l'ajout du signal aux points triples d'un diagramme de stabilité à double boîte quantique [2].	52
4.12	Exemple d'effacements périodiques de certaines parties des lignes de transitions sur un diagramme expérimental (gauche) et sur une simulation (droite), identifiées par des cercles rouges.	53
4.13	Exemples de signaux de transistors mono-électroniques mesurés (haut) et simulés (bas).	54
4.14	Exemple de diagrammes de stabilité utilisés pour l'entraînement et le test du CycleGAN pour l'ajout de bruit expérimental.	58
5.1	Exemples de diagrammes de stabilité simulés (à gauche pour chaque image) et leurs vérités terrains (à droite) générés pour l'entraînement de l'algorithme.	67

5.2 Exemples de diagrammes de stabilité expérimentaux (à gauche pour chaque image) et leurs identifications faites par l'algorithme de segmentation (à droite). À noter que le diagramme du coin inférieur droit est un exemple de diagramme expérimental sans ligne de transition, montrant l'identification où aucune ligne de transition n'est coloriée, mais où tous les pixels de bruit ont été retirés, laissant l'image vide. 69

Introduction

Mise en contexte

La quête du progrès, idée avancée par Descartes dans son *Discours sur la méthode*, n'a jamais été autant d'actualité qu'aujourd'hui, où l'on recherche toujours à faire plus, plus grand et plus rapidement. Cette quête persiste également dans le milieu de l'informatique, où des ordinateurs plus puissants et possédant de plus en plus de transistors sont créés chaque année. Cette croissance respecte d'ailleurs depuis longtemps une loi empirique, nommée « loi de Moore » [3], énonçant que la puissance d'un ordinateur double à chaque année. Cette loi atteindra cependant ses limites sous peu, les processeurs étant constitués de transistors si petits que des effets quantiques peuvent apparaître et nuire aux calculs exécutés par l'ordinateur classique [4].

La quête du progrès n'étant pas arrivée à ses fins, une nouvelle idée d'ordinateur a vu le jour, rendant possible la poursuite de cette augmentation de puissance : l'ordinateur quantique. Utilisant les propriétés de la mécanique quantique et n'étant plus restreint au fameux espace binaire discret (0 ET 1) régissant le fonctionnement des ordinateurs classiques, cet ordinateur pourrait être utilisé pour accélérer, de plusieurs ordres de grandeurs, certains types de calculs et opérations [5] [6]. D'ailleurs, l'engouement vers les technologies quantiques a favorisé la création d'un nouveau domaine, l'informatique quantique, qui trouve déjà plusieurs applications, dans le domaine de la cryptographie [7] ou du développement de nouveaux médicaments [8] par exemple, et démontre déjà une efficacité largement supérieure à celle l'informatique classique.

Pour obtenir un tel gain de puissance, le transistor, unité de calcul fondamentale de l'ordinateur classique, est remplacé par un qubit (bit quantique). Une fois développé et mis à plus grande échelle, un processeur quantique pourrait apporter une puissance de calcul inégalée. En effet, il serait possible, avec seulement cent qubits, d'avoir une puissance de

calcul égale à celle d'un ordinateur constitué d'autant de transistors qu'il y a de particules dans l'univers visible. À ce jour, la course à l'ordinateur quantique se situe, en grande partie, dans l'augmentation du temps de vie et la manipulation (en vue de l'exécution de portes logiques) du qubit. Il existe toutefois de nombreuses avenues de recherche pour la réalisation d'une telle unité, utilisant par exemple des jonctions Josephson (qubits supraconducteurs), des atomes froids (trappes à ions) ou encore le spin des électrons (qubits de spins) [9]. C'est cette dernière architecture qui sera utilisée dans le cadre de ce mémoire.

Qubits de spin

Tout d'abord, le qubit, comme mentionné plus haut, est le composant élémentaire d'un processeur quantique. Tout comme le transistor, son but est d'encoder l'information. Pour y arriver, un système binaire, mais cette fois quantique, doit être utilisé. C'est ce qui motive l'utilisation de l'électron et de son spin $1/2$, remplissant naturellement les exigences du système [10] [11] [12], comme architecture pour la conception d'un qubit.

Les dernières années ont d'ailleurs été particulièrement florissantes pour le domaine et ont apportées des percées significatives. Des temps de cohérences de plusieurs heures [13], des portes logiques universelles et des opérations à un qubit fidèles à plus de 99.9 % [14] ainsi que l'utilisation de techniques de fabrication déjà existantes [15] sont des exemples et avantages qui démontrent que l'architecture des qubits de spin pourrait être la solution de choix pour la conception d'un ordinateur quantique.

Dans le groupe du professeur Pioro-Ladrière, ce sont des dispositifs à boîtes quantiques qui sont utilisés à titre de qubits de spin. Ce type de dispositif, bien que permettant un bon contrôle du système quantique grâce à des grilles de potentiel, amène une problématique, tant au niveau de sa fabrication que de son utilisation, en ce qui concerne son échelonnabilité en vue de la création d'un processeur quantique [16].

Dans le cas présent, c'est plus précisément la problématique de l'initialisation des boîtes quantiques dans le régime à un électron, dans le but de pouvoir les utiliser comme qubit, qui est au coeur du projet. En effet, l'analyse d'une mesure en vue d'une prise de décision n'est pas toujours facile. N'ayant pas d'outil d'analyse performant pour une mesure de boîte quantique, la compréhension de celle-ci ainsi que l'ajustement des grilles de potentiel pour obtenir le régime voulu est un travail colossal nécessitant l'expertise d'un expérimentateur maîtrisant le domaine [17]. Pour résoudre ce problème, la solution apportée consiste en

l'automatisation de l'analyse d'une mesure expérimentale, permettant de faciliter l'implémentation d'un algorithme de prise de décisions.

Dans le cadre de ce travail, une suite d'étapes de simulation et d'augmentation de données ont été implémentées pour entraîner un algorithme d'apprentissage à interpréter les mesures faites en laboratoire, étape cruciale à l'automatisation de l'initialisation d'une boîte quantique dans le régime à un électron. Ce mémoire présente d'abord l'explication des aspects théoriques et expérimentaux concernant les dispositifs et leurs mesures. Ensuite, un cahier des charges sera élaboré afin de présenter les approches déjà explorées pour la résolution de ce problème, leurs limitations, ainsi que la stratégie utilisée pour créer une méthode générale et échelonnable. Seront par la suite décrits le simulateur, les fonctions de traitement d'images et l'algorithme d'apprentissage qu'ils permettent d'entraîner. Finalement, des améliorations, avantages de la technique et avenues possibles de continuité seront présentés.

Chapitre 1

Mesure et initialisation d'une boîte quantique

1.1 Définition de la boîte quantique

Une boîte quantique est une structure de quelques nanomètres, permettant le confinement tridimensionnel des électrons. Si cette structure se retrouve dans des conditions de sorte qu'elle obéisse aux lois de la mécanique quantique, des électrons pourront y être piégés de façon discrète, c'est-à-dire qu'il sera possible de compter un à un les électrons qui seront ajoutés. Pour comprendre le fonctionnement de ce composant, un modèle mathématique simplifié décrivant les grandes lignes de la création d'une boîte quantique sera d'abord présentée, laquelle sera suivie d'une approche expérimentale démontrant ce qui doit être fait en laboratoire pour obtenir ce type de système.

1.1.1 Modèle simplifié

Pour développer une intuition sur ce composant nano-électronique qu'est la boîte quantique, il peut être utile d'utiliser un modèle physique simple, soit le modèle orthodoxe [2][18], pour d'abord comprendre le principe derrière le confinement des électrons. Pour ce faire, un circuit simplifié représentant une boîte contrôlée par une source de potentiel V_g est utilisée telle que schématisée à la figure 1.1.

Dans ce circuit, l'île est donc le point d'intérêt (encadré pointillé sur la figure 1.1), et formera la boîte quantique. Il est à noter que la barrière tunnel séparant l'île de la source permet le couplage et le passage des électrons entre les deux éléments. Considérant d'abord q la charge induite par la source sur l'île ($q = -C_s V_g$) et N le nombre d'électrons en excès

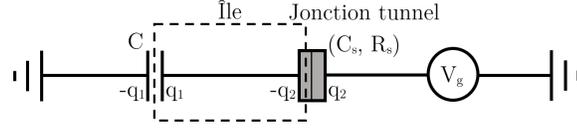


FIGURE 1.1 Circuit électrique décrivant schématiquement une boîte quantique. Ici, la source est séparée de l'île par une jonction tunnel (de capacité C_s et résistance tunnel R_s), q_1 est la charge d'une plaque du condensateur de capacité C , q_2 est la charge d'une plaque du condensateur de capacité C_s , et V_g est le potentiel appliqué à la source.

sur l'île, il sera possible de trouver les charges q_1 et q_2 . En effet, à partir de la différence de potentiel aux bornes des condensateurs C et C_s et de la charge nette sur l'île, les équations 1.1 sont trouvées.

$$q_1 = \frac{eN - q}{1 + C/C_s} \quad q_2 = -\frac{eN - q}{1 + C/C_s} - q \quad (1.1)$$

Connaissant maintenant les charges impliquées dans le système, l'énergie électrostatique sur l'île peut être trouvée, explicitée à l'équation 1.2. Ensuite, grâce à l'énergie de charge, donnée à l'équation 1.3 dans l'hypothèse où l'île était vide au départ, il est possible de résoudre graphiquement le rapport E_{el}/E_c pour différentes valeurs de N (en négligeant le terme $q^2/2C_s$ de l'équation 1.2) afin d'en déduire le niveau d'énergie minimal du système, illustré à la figure 1.2.

$$E_{el} = \frac{q_1^2}{2C} + \frac{q_2^2}{2C_s} + \frac{q_2q}{C_s} = \frac{e^2}{2(C + C_s)} \left(N - \frac{q}{e}\right)^2 - \frac{q^2}{2C_s} \quad (1.2)$$

$$E_c = \frac{e^2}{C_\Sigma} = \frac{N^2}{C + C_s} \quad (1.3)$$

La résolution graphique du rapport E_{el}/E_c montre que la variation de q générée par une modification du potentiel V_g permet de capter ou de retirer des charges sur l'île. En effet, celle-ci « désire » rester dans l'état de moindre énergie (dans l'hypothèse de température nulle, sinon il y aura des fluctuations dues à la température), et modifie son occupation électronique pour le faire. Il est finalement possible d'en déduire le nombre d'électrons en excès se trouvant sur l'île, donné par l'équation 1.4.

$$N = \left\lfloor \frac{q}{e} + \frac{1}{2} \right\rfloor \quad (1.4)$$

En conclusion, une boîte quantique est d'abord un système confiné où les électrons sont

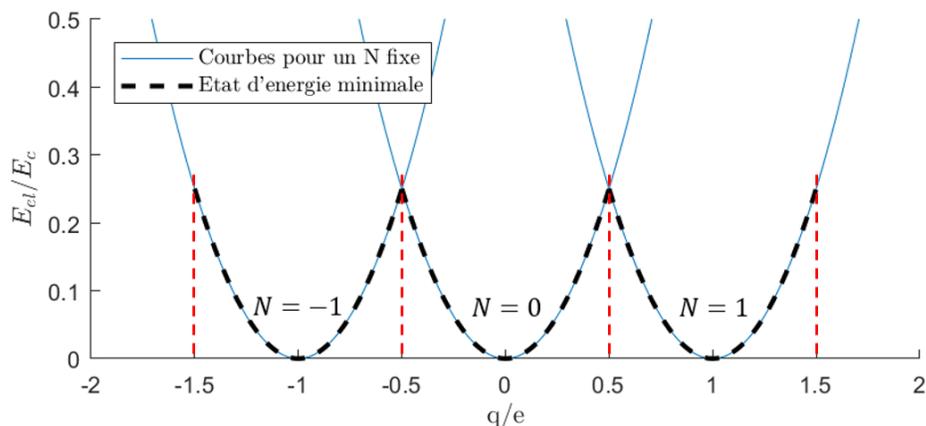


FIGURE 1.2 Représentation graphique des rapports d'énergies E_{el} et E_c pour $N = -1, 0$ et 1 . À noter que le nombre d'électrons à l'intérieur de l'île serait modifié à chaque endroit où il y a une ligne pointillée rouge. Ces lignes arrivent d'ailleurs à chaque valeur demi-entière de q/e , expliquant le terme $1/2$ de l'équation 1.4.

dans un état stable, représenté ici par la séparation entre un condensateur et une jonction tunnel. Le couplage existant entre l'île et la source permet ensuite d'appliquer un potentiel V_g pour modifier l'occupation électronique à l'intérieur de l'île. Il est à noter que, pour discrétiser le nombre d'électrons (en plus d'autres propriétés comme la température, la taille du confinement, etc.), la source de potentiel doit être suffisamment isolée de l'île pour pouvoir confiner et discrétiser la charge à l'intérieur de l'île.

1.1.2 Approche expérimentale

Comme mentionné plus haut, il est possible d'utiliser des méthodes déjà existantes pour la fabrication de dispositifs respectant les caractéristiques décrites lors de l'approche mathématique. En effet, l'une des architectures de qubit de spin utilise la technologie des composants MOS (métal - oxyde - semi-conducteur), possédant tout ce qu'il faut pour confiner un électron et le contrôler [19].

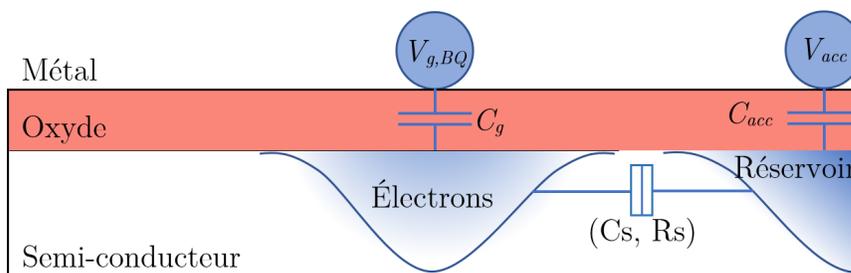


FIGURE 1.3 Schéma d'un dispositif à boîte quantique MOS.

Il s'agit donc, comme illustré à la figure 1.3, de faire un empilement de couches selon l'ordre : une couche métallique, une d'oxyde, et une dernière composée d'un semi-conducteur. La première est métallique et permet d'appliquer un potentiel (V_g). Elle est appelée grille de potentiel, et sert au contrôle de l'occupation électronique dans la boîte quantique. La dernière couche est le semi-conducteur. Ayant une faible densité électronique intrinsèque, elle est essentielle à l'opération dans un régime à faible nombre d'électrons. Finalement, la grille et le semi-conducteur sont séparés par un oxyde (isolant), permettant le couplage capacitif entre la source V_g (grille) et la boîte quantique. À la différence de l'approche mathématique, il est difficile dans ces conditions de transférer des électrons de la source V_g vers la boîte quantique. C'est pourquoi une seconde grille V_{acc} est utilisée afin de créer un deuxième îlot, appelé réservoir, permettant de donner à la boîte quantique accès à un certain nombre d'électrons. De plus, le milieu semi-conducteur crée naturellement la jonction tunnel entre les îlots, permettant le confinement et la discrétisation de la charge.

Un premier confinement est donc créé par l'utilisation du composant MOS, ayant naturellement les propriétés électroniques attendues pour un gaz d'électrons bidimensionnel. Pour piéger des électrons, un dernier confinement est nécessaire, atteint par la combinaison de potentiels appliqués sur les grilles, disposées selon des géométries particulières pour faciliter la création de la boîte quantique. De cette façon, si la région confinée est suffisamment petite, elle obéira aux propriétés de la mécanique quantique et pourra être utilisée dans le régime du qubit.

1.2 Initialisation de la boîte quantique dans le régime du qubit

L'initialisation d'un dispositif à boîte quantique, c'est à dire la tâche de trouver les potentiels à appliquer sur les grilles (par exemple les grilles de potentiel du transistor mono-électronique et de la boîte quantique, tel que décrit dans les dispositifs jusqu'ici) pour ne retrouver qu'un seul électron à l'intérieur de la boîte, nécessite l'emploi d'un instrument de mesure capable de détecter une variation d'une seule charge. Pour ce faire, deux grandes méthodes peuvent être utilisées, soit la mesure en transport [18], ou la mesure en détection de charges [2]. Dans le groupe du professeur Pioro-Ladrière, c'est cette dernière méthode qui est utilisée, réalisée grâce à un type particulier de transistor : le transistor mono-électronique[20].

1.2.1 Transistor mono-électronique et détection de charges

Le TME (transistor mono-électronique) a l'avantage de pouvoir être fabriqué sur un composant MOS, la même technologie que celle utilisée pour la fabrication des boîtes quantiques. En effet, il s'agit encore une fois d'un îlot d'électrons, mais cette fois avec un grand nombre de charges. En plus du dispositif décrit plus haut, deux réservoirs d'électrons, situés de part et d'autre de l'îlot, sont créés grâce à des grilles d'accumulation, et sont appelés *source* et *drain* (voir figure 1.4). Ceux-ci seront utilisés pour faire circuler un courant partant de la source et allant jusqu'au drain, en passant par l'îlot.

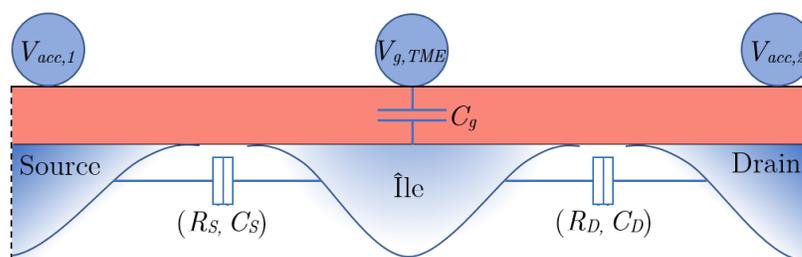


FIGURE 1.4 Schéma d'un dispositif à boîte quantique MOS.

C'est ce passage par l'îlot qui fait gagner tout l'intérêt à ce dispositif, étant régi par le blocage de Coulomb [21]. En effet, étant donné une occupation électronique restreinte et quantifiée à l'intérieur de l'îlot, il sera possible d'observer des pics de conductance maximale et des plateaux de conductance nulle dans le comportement IV du transistor [2]. Le signal abrupte attendu est cependant adouci, en pratique, dus au grand nombre d'électrons dans les îlots, aux effets thermiques, aux interactions avec les phonons, au bruit de charges et au couplage tunnel avec les réservoirs, donnant un caractère oscillatoire à la courbe IV mesurée.

La conductance du transistor, étant caractérisée par le transport d'un électron à la fois, est donc extrêmement sensible à l'environnement électrostatique de l'îlot. En effet, tout ajout ou retrait de charge ayant un couplage capacitif avec l'îlot agit comme une modification du potentiel d'une grille couplée : le comportement sinusoïdal attendu subit un retard de phase, d'un ΔV équivalent au potentiel effectif que la grille doit fournir pour lutter contre la nouvelle charge. C'est ce qui en fait l'outil de mesure idéal pour une boîte quantique, pouvant détecter toute modification de son occupation électronique. La figure 1.5 montre un schéma du courant que peut mesurer un TME couplé à une boîte quantique, et l'impact de l'ajout d'une charge dans celle-ci.

La détection de charges consiste donc en l'utilisation d'un transistor mono-électronique, placé à proximité de la boîte quantique, y mesurant toute modification de sa charge interne. Un couplage capacitif existe alors entre le transistor et la boîte, et l'ajout d'un électron par

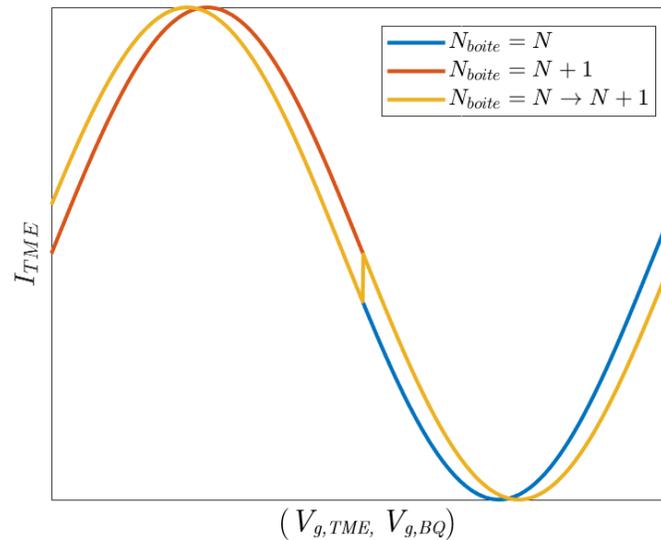


FIGURE 1.5 Schéma de la courbe IV d'un transistor mono-électronique faisant la mesure d'une boîte quantique pour différentes occupations[1]. La courbe rouge peut s'obtenir grâce à l'équation $I_{TME} = \sin(V_g)$, la bleue grâce à $I_{TME} = \sin(V_g + \phi)$ où ϕ est un retour de phase, et la jaune avec $I_{TME} = \sin(V_g + \Phi)$ où $\Phi = \phi \times$ une fonction escalier

l'augmentation du potentiel V_g cause une modification de la conductance du dit transistor, pouvant être identifiée par différents traitements mathématiques.

1.2.2 Mesure d'un diagramme de stabilité

Pour trouver l'occupation électronique au sein d'une boîte quantique, il s'agit donc de modifier les potentiels appliqués sur les grilles du dispositif, de mesurer le signal provenant du transistor mono-électronique, et d'identifier les potentiels pour lesquels il semble y avoir un retour de phase 1.5.

Pour explorer plus d'un balayage en potentiel sur plus d'une grille à la fois, c'est plutôt une carte à deux dimensions, appelée *diagramme de stabilité*, qui est faite en balayant deux grilles simultanément. Le même signal qu'expliqué précédemment est mesuré, puis l'image est transformée en base deux grâce à une analyse mathématique [1]. Ce dernier algorithme a pour but, à partir d'une mesure en détection de charges, d'identifier les pixels pour lesquels il y a eu une modification de l'occupation électronique, de les marquer par le chiffre un, et de remplacer tous les autres par des zéros. La figure 1.6 montre un schéma du courant que peut mesurer un TME couplé à une boîte quantique pendant la mesure d'un diagramme de

stabilité.

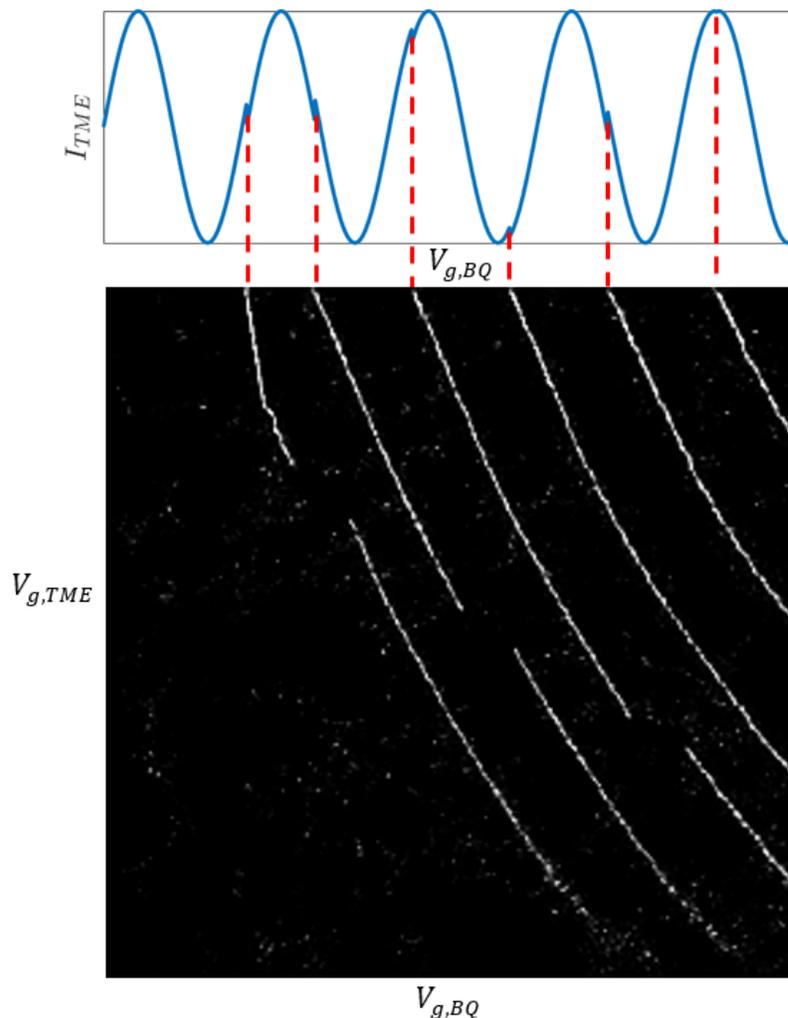


FIGURE 1.6 Schéma de la courbe IV d'un transistor mono-électronique faisant la mesure d'une boîte quantique ainsi qu'un exemple de diagramme de stabilité lui étant associé (simulé). Les lignes rouge pointillées lient le retour de phase dans le courant du TME à la transition électronique associée.

Il est à noter que le système est physique, et que les potentiels balayés ont une échelle continue et linéaire, ne pouvant donc causer aucune discontinuité sur le diagramme de stabilité. Un pixel étant identifié comme une modification de l'occupation électronique, pour un potentiel donné, fait donc généralement partie d'une ligne complète, appelée ligne de transition.

L'identification des lignes de transition du diagramme de stabilité peut finalement être utilisée pour initialiser une boîte quantique dans le régime à un électron. Il suffit

effectivement de trouver quelles valeurs de potentiels génèrent la première ligne de transition. En effet, si le nombre total d'électron dans la boîte tombe à zéro, il est impossible d'en retirer un autre par la diminution V_g , permettant ainsi de trouver une **première** ligne de transition. Celle-ci indiquera que l'occupation électronique dans la boîte est forcément d'un seul électron, et que le régime désiré est atteint.

Chapitre 2

Cahier des charges

Le chapitre précédent présente les étapes et concepts permettant la mesure d'un diagramme de stabilité. Son rôle dans l'initialisation d'un dispositif à boîte quantique dans le régime à un électron a également été décrit. Un problème est cependant rapidement rencontré, en ce qui concerne l'identification des lignes de transition sur le diagramme de stabilité. En effet, il n'existe encore aucun algorithme pouvant faire abstraction de tous les bruits et défauts que peuvent engendrer une mesure expérimentale, empêchant ainsi à un ordinateur de déterminer avec grande fiabilité l'occupation électronique dans un dispositif.

Cette identification automatique est cependant un critère clé à l'échelonnabilité des qubits de spin, permettant d'effectuer l'initialisation par une machine, et fait partie de la présente étude. Ce chapitre présente donc d'abord en quoi il est difficile d'identifier une simple ligne dans une image, ainsi que les méthodes proposées jusqu'à présent. Il présentera par la suite les objectifs et limitations de l'algorithme à développer, ainsi que le plan détaillé d'une nouvelle méthode d'identification.

2.1 Problématique

Le problème est simple : la mesure se fait dans la vie réelle. Tout ce qui a été expliqué et simplifié jusqu'à présent peut en fait devenir une cause d'erreur, pouvant modifier, ajouter ou effacer des pixels et des lignes de transition sur le diagramme de stabilité. En effet, le fait d'avoir un transistor qui détecte la variation d'une seule charge le rend sensible aux défauts dans le maillage du réseau des atomes et aux atomes indésirables présents à proximité. De plus, la température non-nulle affecte l'occupation des niveaux d'énergie, et

de nombreux passages tunnels et autres effets quantiques avec les matériaux ne sont pas expliqués ou pris en considération. Qui plus est, les instruments de mesure et les traitements mathématiques peuvent induire davantage d'erreurs et modifier les résultats qui étaient attendus si des paramètres sont mal ajustés, ou changent d'un dispositif à l'autre. Finalement, cette succession de défauts résulte en un diagramme de stabilité où les lignes peuvent se briser, se courber, s'effacer ou s'évaser, et où chaque pixel identifié comme une transition n'en est pas forcément une (voir exemple à la figure 1.6).

La question est donc : comment identifier une ligne qui n'en est pas une ? Deux grandes méthodes ont jusqu'ici été utilisées dans la communauté, comportant chacune leurs avantages. Elles ne peuvent toutefois réussir, de façon robuste et générale, à identifier et différencier chacune des lignes de transition d'un diagramme de stabilité. La prochaine section vise à présenter ces deux méthodes et à expliquer les raisons pour lesquelles elles limitent l'échelonnabilité des qubits de spin.

2.2 Travaux antérieurs

2.2.1 Recherche de formes connues

Des groupes comme celui de Vandersypen à Delft [22] se sont déjà penchés sur le problème et ont opté pour des méthodes de reconnaissance de formes. Ces travaux ont mené à des résultats prometteurs, permettant de reconnaître efficacement le signal voulu pour initialiser la boîte quantique.

Cette technique, développée à Delft, consiste en un algorithme qui compare une forme attendue à la mesure expérimentale. La région ainsi délimitée sur le diagramme de stabilité est identifiée grâce à la forme placée. Puisque cette forme est représentative de l'occupation électronique du dispositif, il est possible de déterminer le régime du qubit sur le diagramme de stabilité. Cette méthode a d'ailleurs fait ses preuves en automatisant l'initialisation de deux boîtes quantiques dans le régime à un électron. Un seul problème persiste lorsque la mesure ne ressemble pas tout à fait à ce qui était attendu. La recherche de forme et l'identification qui en est faite peuvent s'en retrouver biaisées, donnant une incertitude sur l'initialisation qui en résulte.

2.2.2 Recherche de lignes

Une deuxième méthode, quant à elle, consiste à identifier directement les lignes sur une image, grâce à des techniques de reconnaissance comme la transformée de Hough [23], la transformée de Hough modifiée [24], ou EDline [25].

Les défis, imposés par la technique, résident encore une fois dans le cas où la mesure est perturbée par des effets expérimentaux imprévus. Il peut ainsi s'ajouter une identification biaisée d'un artefact expérimental, ou encore une non-identification d'une ligne de transition, causant une erreur de mesure. C'est d'ailleurs un des buts de la suite du projet : pouvoir faire abstraction de ces erreurs de mesure grâce à l'apprentissage statistique.

La stratégie, développée ici à Sherbrooke [26], est donc d'avoir optimisé un algorithme de traitement de signal pour améliorer la recherche de transitions dans la courbe IV du transistor mono-électronique. Cet algorithme permet, à défaut d'ajouter un fond plus bruité que les traitements déjà existant, de maximiser le nombre de pixels composant les lignes de transition. Par la suite, la transformée de Hough modifiée (première phase du projet implémentée par Maxime Lapointe-Major), ou encore EDline (amélioration apportée par Olivier Germain, étudiant à la maîtrise en mathématique dans le groupe du professeur Félix Camirand Lemyre) permettent d'identifier les lignes de transitions sur le diagramme de stabilité. Un des avantages de ces algorithmes est leur robustesse face au fond bruité pour la reconnaissance de lignes. La maximisation du nombre de pixels par ligne, grâce au pré-traitement de signal, permet donc de bien identifier et distinguer les lignes de transitions sur le diagramme de stabilité, et d'ainsi faire l'initialisation de la boîte quantique dans le régime du qubit.

Cette stratégie a également fait ses preuves, montrant l'initialisation d'une boîte quantique dans le régime à un électron grâce aux techniques implémentées par Maxime Lapointe-Major. De plus, les bonifications apportées par Olivier Germain permettent d'augmenter la robustesse et la rapidité de recherche des lignes. Finalement, il est à noter que le développement de ces techniques, étant maintenant un projet fondé IQ, sont encore sur la table et se verront bonifiées par des techniques d'apprentissage statistique apportées par le professeur Félix Camirand Lemyre, permettant d'augmenter la rapidité et l'efficacité de l'initialisation.

2.2.3 Techniques d'apprentissage

Une nouvelle méthode [27] montre qu'il est possible d'utiliser les techniques d'apprentissage récemment développées pour aider à l'initialisation des boîtes quantiques. En effet, les

réseaux de neurones sont des outils de plus en plus performants, aujourd'hui appliqués dans plus en plus de domaines. Ces réseaux sont d'ailleurs utiles en reconnaissance d'images, et peuvent être utilisés pour la reconnaissance de régimes d'opérations de boîtes quantiques.

La technique consiste à faire apprendre à un algorithme la différence visuelle qu'il existe entre les régimes qu'il est possible d'observer dans le système. Il existe, en fait, des éléments visuels très distinctifs entre un dispositif ne possédant aucune boîte quantique, en possédant une, deux, etc. Il s'agit donc d'avoir une grande quantité de ses diagrammes de stabilité distinctifs, et d'entraîner un réseau de neurones à les différencier. Pour ce faire, une simple valeur entière est attribuée à chaque diagramme de stabilité, constituant ainsi la base de données d'entraînement de l'algorithme. Pour en vérifier les performances, l'article démontre qu'il a été possible d'identifier, avec succès, le régime (nombre de boîtes quantiques en opération) sur des diagrammes de stabilité connus des chercheurs, mais non de l'algorithme. De plus, et ce qui représente le résultat le plus éloquent, est que l'algorithme d'apprentissage a pu être entraîné à partir de données simulées uniquement, puis s'est avéré fonctionnel aussi sur des diagrammes de stabilité expérimentaux. Cette façon de faire, bien que n'identifiant que le nombre de boîtes quantiques en opération et non le nombre d'électrons qu'elles possèdent, semble être très prometteuse pour le domaine.

En conclusion, cette technique s'est avérée efficace pour reconnaître le régime dans lequel le système se trouve. Une limitation s'impose cependant en ce qui concerne la prédiction de l'algorithme en cas de données bruitées. En effet, l'algorithme d'apprentissage est entraîné grâce à une grande banque de diagrammes de stabilité simulés par le modèle physique de Thomas-Fermi 1D. Ce choix restreint donc l'apprentissage de l'algorithme à ce que le modèle physique peut simuler. La présence de bruit un diagramme expérimental peut donc nuire à la performance de l'algorithme d'apprentissage. La probabilité calculée d'être dans l'un ou l'autre des régimes peut finalement être biaisée, et est difficilement vérifiable sans identification géométrique faite directement sur le diagramme de stabilité.

2.2.4 Combinaison de reconnaissance de formes et d'algorithme d'apprentissage

Une dernière méthode [28], consistant en une combinaison de reconnaissance géométrique et de techniques d'apprentissage, a montré l'initialisation de deux boîtes quantiques dans le régime à un électron. L'avancée de cette technique est qu'elle a pour finalité l'identification d'une transition électronique à l'aide d'un algorithme d'apprentissage, et non seulement la reconnaissance du nombre de boîtes quantiques. La stratégie est donc d'utiliser, à titre d'entraînement, des diagrammes de stabilité en plus des informations extraites par

une transformée de Hough¹[23] identifiant préalablement les lignes de transitions. Ces informations sont ensuite utilisées pour prédire l'emplacement du régime d'intérêt sur l'image.

En conclusion, il s'agit de la première technique utilisant des techniques d'apprentissage pour trouver le nombre d'électrons dans une boîte quantique. Ces avancées sont donc très importantes pour l'échelonnabilité de la technologie, montrant que l'initialisation peut être faite par intelligence artificielle. Les défis pour le futur de cette méthode résident donc dans l'augmentation de la robustesse face aux différents bruits expérimentaux. En effet, comme mentionné dans les sections sur 2.2.2 et 2.2.3, ces techniques sont limitées par la présence d'artefacts imprévus, ou non simulés par le modèle physique servant à la création de la base de données d'entraînement.

2.3 Méthode proposée

Comme il a été possible de le voir, les méthodes existantes sont toutes prometteuses en ce qui concerne l'initialisation de la boîte quantique dans le régime du qubit, mais se retrouvent quelque peu limitées par la présence de bruit expérimental. C'est donc la problématique qui sera tentée d'être résolue dans ce travail : pouvoir identifier et distinguer les lignes de transition sur un diagramme de stabilité, et ce, même en présence d'artefacts expérimentaux.

Pour ce faire, un algorithme d'apprentissage de segmentation d'image sera utilisé. L'emploi d'un entraînement entièrement simulé est également favorisé, nécessitant l'implémentation d'un générateur de diagrammes de stabilité fidèles à la réalité pour la création de la base de données. La méthode développée consistera, premièrement, en un simulateur de diagrammes de stabilité à partir d'un modèle physique, deuxièmement, en un algorithme d'augmentation de données permettant la génération d'artefacts expérimentaux, et, troisièmement, en un algorithme d'apprentissage profond de segmentation de données pour l'identification de transitions électroniques sur diagrammes expérimentaux. Les sous-sections suivantes décrivent les objectifs et restrictions de la méthode à développer.

1. En quelques mots, cet algorithme permet de trouver les segments de lignes dans une image grâce à un système de vote entre les pixel et leur valeur. Une fois qu'une suite de pixel est identifié selon l'équation voulue (une droite ici), le segment est enregistré, et la recherche se poursuit jusqu'à ce que l'image au complet ait été analysée.

2.3.1 Objectifs et restrictions

Les outils développés visent une identification pixel à pixel d'un diagramme de stabilité, où les lignes de transition différentes se verraient attribuer une valeur de pixel différente. Une recherche d'une valeur de pixel en particulier informerait sur la position de la ligne de transition entière étant identifiée à cette valeur, et permettrait la prise de décision pour un algorithme d'initialisation. De plus, l'emploi de techniques d'augmentation de données rendrait l'algorithme robuste aux artefacts expérimentaux sans avoir besoin de modèles physiques lourds et complexes pouvant interpréter et calculer fidèlement la réalité du qubit de spin. Ainsi, le module d'identification posséderait les avantages du gain d'information et de simplicité des techniques de reconnaissance de formes en plus de ceux de la généralité et de la robustesse des techniques d'apprentissage, tout en n'ayant aucun de leurs désavantages.

Le module doit requérir un minimum de paramètres physiques. Ceux-ci consisteront, entre autres, à la création du dispositif par éléments finis, au positionnement des grilles de potentiel, aux potentiels leurs étant appliqués et aux paramètres du modèle choisi. Ils devront faciliter la génération du type de diagramme désiré, permettant de créer aisément la base de données d'entraînement. Il en est de même pour l'algorithme d'augmentation de données, où une modification d'un diagramme pour l'ajout d'un artefact expérimental doit demander le moins de paramètres et de temps de calcul possible.

L'algorithme développé se veut général, pouvant être utilisé pour différents types de dispositifs et régimes d'opérations. Afin de démontrer la généralité de l'identification en ce qui concerne le dispositif, un modèle physique à une seule dimension sera utilisé pour identifier les transitions de diagrammes de stabilité acquis sur des dispositifs MOS. Pour ce qui est du régime, seul celui à une boîte quantique sera simulé et testé, mais sans contraintes sur le nombre de transitions observées.

Le dernier critère à respecter est la robustesse face à différentes observations expérimentales qui ne peuvent être simulées par le modèle physique, acquise par l'augmentation de données. Afin de démontrer qu'il est bien possible d'atteindre cette performance, une grande variété de diagrammes possédant ces artefacts seront utilisés à titre de test.

Finalement, il est à noter que ce travail vise une preuve de concept d'une nouvelle méthode d'identification des lignes de transition. Le but n'est donc pas la performance, mais de faire la preuve de concept qu'il est possible de réduire un lourd problème quantique en une série d'étapes de modification de pixels.

2.3.2 Justification et philosophie de l'algorithme

Il va de soi que l'idée de modifier manuellement des pixels afin de créer, au bon vouloir, une image qui semble être plus près de la réalité, et d'utiliser cette image comme entraînement d'une intelligence artificielle peut, à première vue, sembler un peu simpliste pour régler tous les problèmes mentionnés plus haut. Voici, en quelques mots, l'idée derrière l'implémentation de cette technique, et pourquoi elle devrait pouvoir fonctionner.

Il y a, de façon générale, deux concepts permettant d'identifier un signal voulu dans une mesure :

- La compréhension du bruit : si le comportement du bruit est bien connu, il suffira de le retirer des données ou de le traiter. Des techniques de filtrage peuvent par exemple être employées pour y arriver. Ce qui en résultera est le signal d'intérêt.
- La compréhension du signal : si le comportement du signal est bien connu, il suffira de l'extraire directement grâce à des méthodes de lissage par exemple.

En ce qui concerne les diagrammes de stabilité, enlever le bruit serait une tâche excessivement complexe, pouvant venir de multiples causes et traitements. C'est donc la compréhension du signal qui est visée.

Avant la venue des techniques d'apprentissage, ce sont généralement des équations (par exemple la régression linéaire) qui étaient utilisées pour extraire le signal d'une mesure. Cependant, l'arrivée de l'apprentissage non-supervisé permet à des réseaux de neurones de comprendre la réalité en leur permettant, en quelque sorte, d'instaurer leur propre modèle physique. C'est ce qui leur donne leur grande généralité, étant capable (ou limités...) de faire un apprentissage avec les données qui leur seront fournies.

En d'autres mots, un modèle d'apprentissage sera en mesure de comprendre la réalité si les données qui lui sont fournies en entraînement possèdent les éléments qui composent cette réalité. Il est cependant à noter que ces éléments sont limités à ce que la mesure permet d'observer, et non au système physique complet qui est mesuré. Je crois donc que ce qui est primordial pour l'algorithme d'apprentissage qui façonnera lui-même son modèle, ce n'est pas la façon dont ces éléments arrivent sur la mesure, mais la présence de tous les éléments constituant la réalité. Une illustration de ce principe pourrait être le problème de distinction entre un chat et un humain dans une photo. Est-il nécessaire de comprendre le système biologique complexe des deux êtres pour pouvoir les différencier, ou suffit-il de savoir que l'un marche sur deux jambes et a de la peau visible, tandis que l'autre marche sur quatre pattes et est poilu sur l'entièreté de son corps ? Donc, si la distinction d'éléments simples et caractéristiques d'un objet complexe peut en permettre l'identification, il en est de même pour les lignes de transition.

L'idée est donc la suivante : capitaliser sur le fait que le système quantique cryogénique complexe de la boîte quantique et son initialisation dans le régime du qubit de spin utilisent une mesure aussi simple que celle d'une ligne. Bien que ce type de système soit difficile à simuler avec précision, il est facile d'interpréter une ligne de transition, et d'attribuer les artefacts visuels observés à leurs causes expérimentales. Dans la réalité, ces artefacts sont souvent en lien les uns avec les autres et proviennent d'interactions complexes, d'où le fait qu'il soit difficile d'en faire une simulation adéquate. C'est à ce moment que le concept d'utiliser une variété statistique plus grande arrive. En effet, rendre indépendants les artefacts et les dessiner indépendamment sur un diagramme de stabilité prend nécessairement en compte le sous-ensemble statistique réel où ces éléments avaient des liens entre eux. Ceci permettrait donc à l'algorithme d'apprentissage d'avoir connaissance des défauts observables en laboratoire, et de les identifier sur une véritable mesure.

En conclusion, l'algorithme d'identification de lignes de transition vise une compréhension profonde du signal qu'est une ligne de transition et de ses artefacts possibles, pouvant faire abstraction des pixels bruyants. L'apprentissage se fera donc par à un modèle physique pour la génération arbitraire des diagrammes de stabilité, augmentés en un ensemble statistique plus grand que celui de la réalité grâce et des algorithmes de traitement d'images indépendants. La preuve de concept de cette méthode serait une toute nouvelle technique de génération de base de données (dont la validité sera assurée par la réussite de l'identification de mesures expérimentales grâce à l'algorithme entraîné uniquement à partir de cette base simulée), la démonstration d'un outil performant de détection de lignes de transition, ainsi qu'une simplification du problème d'identification à l'implémentation de quelques fonctions de traitement d'images.

Chapitre 3

Implémentation du simulateur

Pour pouvoir être entraîné, un algorithme d'apprentissage nécessite un grand nombre de données d'entraînement. De plus, la tâche à réaliser requiert une identification précise de diagrammes de stabilité expérimentaux. En raison du temps d'acquisition d'une mesure réelle et de la difficulté à l'identifier correctement, l'utilisation d'un simulateur a été préconisée pour créer ces données d'entraînement.

Un modèle physique simple a été utilisé, permettant d'incorporer un certain nombre d'éléments physiques et une facilité d'utilisation tout en gardant une grande rapidité de calcul. Il s'agit du modèle de Thomas-Fermi¹ sur un dispositif 1D, où les électrons se comportent comme un gaz électronique dans un 2DEG (densité d'état constante). Les sections de ce chapitre et l'organigramme présenté à la fin de celui-ci sont dédiés à l'explication de la simulation d'un diagramme de stabilité.

1. Ce modèle, ayant déjà été utilisé dans [27] pour entraîner un réseau de neurones à reconnaître le régime d'un système de boîtes quantique, semblait un bon candidat. En effet, il semble pouvoir reproduire certaines caractéristiques nécessaires à l'entraînement d'un algorithme d'apprentissage. De plus, il permet l'implémentation d'un simulateur rapide et modulable, idéal pour la génération d'une grande quantité de diagrammes de stabilité.

3.1 Profil de potentiel électrostatique

La première étape consiste à créer des îlots d'électrons dans le dispositif. Pour ce faire, des grilles de potentiel sont posées à la surface de celui-ci. En y appliquant un potentiel, un profil de potentiel électrostatique est généré dans le dispositif, permettant d'attirer ou de repousser des électrons.

Ces grilles sont considérées comme des conducteurs cylindriques chargés², centrées et perpendiculaires à l'axe 1D du dispositif (figure 3.1). Elles sont définies par :

- un potentiel appliqué (V_0);
- une position sur le dispositif (x_0);
- une longueur (l);
- un rayon (R_g);
- une distance les séparant des électrons (r);
- une distance «infinie», où le potentiel est considéré nul ($cond_{inf}$).

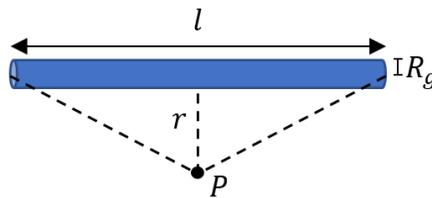


FIGURE 3.1 Schéma d'une grille de potentiel. La direction x dans cette figure, et utilisé dans le reste des équations, sort de la page (⊙)

L'équation analytique 3.1 permet le calcul du profil de potentiel au niveau des électrons dans le dispositif [29] :

$$V_g(r) = \frac{V_0}{r} \cdot \log \left(\frac{r}{l\sqrt{4r^2 + l^2} + l} \right) + k = V_0 \cdot f(r) \quad (3.1)$$

où k est une constante provenant de l'intégration du champ électrique d'un cylindre chargé et r la distance entre le point de calcul et le centre de la grille. Pour les algorithmes, la fonction $V_0 \cdot f(r)$ est gardée en mémoire dans une structure nommée *paramètres_grilles*

2. Justification : les conducteurs chargés permettent d'appliquer des potentiels de façon uniforme, et jouent le rôle de grille de potentiel. En ce qui concerne la géométrie choisie, il s'agit davantage d'une simplification du problème grâce à la symétrie des cylindres, permettant l'implémentation simple et rapide d'un simulateur modulable pour l'utilisateur.

3.2 Profil de densité électronique

Le profil de potentiel généré peut être vu comme la quantité d'énergie, en chaque point du dispositif, disponible pour l'attraction ou la répulsion d'un électron. Il est donc possible de calculer le profil de densité électronique généré par les grilles de potentiel [30] grâce à l'intégrale de Fermi-Dirac (éq. 3.2 [27]),

$$\eta(x) = \int_{\epsilon'_0}^{\infty} \frac{g_0}{1 + e^{\beta(\epsilon - E_F)}} d\epsilon \quad (3.2)$$

où g_0 est la densité d'état, $\beta = 1/kT$, E_F est l'énergie de Fermi et ϵ'_0 est le minimum de bande de conduction modifié.

Dans le calcul de la densité électronique, le minimum de bande de conduction modifié permet de prendre en compte les interactions électrons-électrons. En effet, le profil de potentiel calculé plus tôt est modifié par l'ajout de charges dans le dispositif, modifiant également la quantité d'énergie disponible. Cette quantité est appelée minimum de bande de conduction modifié (éq. 3.3 [27]),

$$\epsilon'_0(x) = \epsilon_0 - V_g(x) + \int K(x, x') \eta(x') dx' \quad (3.3)$$

où ϵ_0 est le minimum de bande de conduction intrinsèque du dispositif, et le terme $K(x, x')$ représente les interactions électroniques (éq. 3.4 [27]),

$$K(x, x') = K_0 / \sqrt{(x - x')^2 + \sigma^2}, \quad (3.4)$$

avec K_0 l'amplitude de l'interaction électronique et σ une petite valeur empêchant la divergence en $x = x'$.

La calcul de la densité électronique se fait ensuite jusqu'à convergence des équations 3.2 et 3.3. Il est à noter que l'amplitude maximale de l'interaction sur le dispositif est donné par K_0/σ . Ces paramètres jouent donc un rôle important pour la convergence de l'algorithme, pouvant même la compromettre sous certaines combinaisons. L'algorithme exécutant ce calcul est présenté à l'algorithme 1.

Algorithme 1 : Calcul de la densité électronique

Entrées : interactions_poids, min_conduction, maille_dispo, dx_maille, V, info_int
Sorties : densité_élec, n_îlots, pos_îlots, N_élec, min_conduction_mod, interaction_élec

- 1 Initialisation des variables ;
- 2 interaction_élec⁽⁰⁾ ← [0...] // taille ← taille(maille_dispo) ;
- 3 densité_élec⁽⁰⁾ ← 5 · V_{normalis} ; // N_élec⁽⁰⁾ ← 5 si V > 0, 0 sinon ;
- 4 [N_élec⁽⁰⁾, n_îlots⁽⁰⁾, pos_îlots⁽⁰⁾] ← calcul_nbe(densité_élec⁽⁰⁾) ;
- 5 Calcul de la densité électronique jusqu'à convergence ;
- 6 **tant que** N_élec⁽ⁿ⁾ ≠ N_élec⁽ⁿ⁺¹⁾ ET pos_îlots⁽ⁿ⁾ ≠ pos_îlots⁽ⁿ⁺¹⁾ **faire**
- 7 | Calcul de la nouvelle densité électronique ;
- 8 | interaction_élec⁽ⁿ⁺¹⁾ ← somme(interactions_poids · densité_élec⁽ⁿ⁾) · dx_maille ;
- 9 | min_conduction_mod⁽ⁿ⁺¹⁾ ← min_conduction - V + interaction_élec⁽ⁿ⁺¹⁾ ;
- 10 | densité_élec⁽ⁿ⁺¹⁾ ← équation 3.2 avec min_conduction_mod⁽ⁿ⁺¹⁾ ;
- 11 | Calcul du nombre d'électrons et de leur position ;
- 12 | [N_élec⁽ⁿ⁺¹⁾, n_îlots⁽ⁿ⁺¹⁾, pos_îlots⁽ⁿ⁺¹⁾] ← calcul_nbe(densité_élec⁽ⁿ⁺¹⁾) ;
- 13 **fin**

3.3 Distribution aléatoire de charges

Un isolant se trouve entre l'interface où les charges peuvent s'accumuler et les grilles de potentiel. Dans la réalité, celui-ci peut contenir des charges fixes, générant elles aussi un champ électrique, jouant un rôle dans l'accumulation des charges.

Pour les simulations, une « carte » aléatoire de défauts est générée selon une densité donnée, nommée η_{oxyde} . Cette carte informe si une charge fixe est présente, ou non, dans un quadrillage entre les grilles et l'interface d'accumulation (voir l'algorithme 2). Le potentiel généré par ces charges peut être calculé via le terme d'interactions électronique de l'équation 3.3, en remplaçant σ par la distance entre la charge et l'interface d'accumulation ($d_{charges-interface}$). Ce profil est nommé V_{oxyde} , et est additionné au potentiel généré par les grilles pour créer le profil de potentiel total, V , lequel est utilisé pour le calcul de l'occupation électronique du dispositif (référer à la ligne 5 de l'algorithme 4).

Algorithme 2 : Calcul du profil de potentiel de la distribution de charges aléatoire

Entrées : maille_dispo, dx_maille, hauteur_grilles, η_{oxyde} , σ , K_0

Sorties : V_{oxyde}

```

1 Calcul du nombre d'électrons présents dans l'oxyde ;
2 nb_électrons ← arrondir(taille(x) · dx_maille · hauteur_grilles ·  $\eta_{\text{oxyde}}$ ) ;

3 Calcul de la distribution aléatoire de charges fixes ;
4 carte_charges ← [0, ..., 0] // taille ← [hauteur_grilles/dx_maille X taille(x)] ;
5 indices_électrons ← « nb_électrons » entiers ∈ [1, taille(carte_charges)] ;
6 carte_charges(indices_électrons) ← 1 ;

7 Calcul du profil de potentiel total ;
8 pour  $i$  ← 1 à nombre de ligne(carte_charges) faire
9   Calcul du poids des interactions électroniques via l'équation 3.4 ;
10   $\sigma$  ←  $\sigma + d_{\text{charges-interface}}$  ;
11  interactions_poids ← équation 3.4 ;

12  Calcul du profil de potentiel pour cette ligne ;
13   $V_{\text{ligne}}$  ← somme(interactions_poids) ; // éléments finis : intégrale ←
    somme ;

14  Calcul du profil de potentiel effectif ;
15   $V_{\text{oxyde}}$  ←  $V_{\text{oxyde}} + V_{\text{ligne}}$  ;
16 fin

```

3.4 Calcul de l'occupation électronique

Une fois la densité électronique calculée, il faut trouver, s'il y a lieu, les emplacements des îlots d'électrons créés ainsi que leurs nombres d'électrons. Pour déterminer s'il y a un îlot, les deux critères suivants doivent être observés sur la densité électronique :

1. $\eta(\text{îlot}) > 0$
2. $\eta(\text{îlot} \pm dx) = 0$

Ensuite, par la minimisation de l'énergie, le nombre d'électrons dans chaque îlot est calculé en arrondissant, à la baisse, la densité électronique composant l'îlot (algorithme 3).

Algorithme 3 : Calcul du nombre d'électrons et de leur position sur le dispositif

Entrées : densité_élec

Sorties : n_îlots, position_îlots, N_élec

- 1 *Calcul de la position et du nombre d'îlots ;*
 - 2 booléen_densité \leftarrow 1 si densité_élec $>$ 0, 0 sinon ;
 - 3 débuts_îlots \leftarrow trouve(*diff*(booléen_densité) = +1) ;
 - 4 fins_îlots \leftarrow trouve(*diff*(booléen_densité) = -1) ;
 - 5 position_îlots \leftarrow [débuts_îlots, fins_îlots] ;
 - 6 n_îlots \leftarrow *taille*(débuts_îlots) ;
 - 7 *Calcul du nombre total d'électrons ;*
 - 8 N_élec \leftarrow arrondir à la baisse(*somme*(densité_élec(position_îlots)))
-

Ce nombre d'électrons par îlot représente l'occupation électronique sur le dispositif. Le calcul de cette occupation pour différentes combinaisons de potentiels de grilles permet le calcul d'un diagramme de stabilité (voir l'algorithme 4), où la quantité mesurée est le nombre d'électrons total accumulés sur le dispositif, étant donné différents potentiels de grilles (exemple à la figure 4.1).

Algorithme 4 : Calcul de l'occupation électronique

Entrées : paramètres_grilles, V_grille1, V_grille2, V_oxyde, interactions_poids, min_conduction, maille_dispo, dx_maille, info_int

Sorties : occupation_élec

```

1 Calcul indépendant des éléments de occupation_élec;
2 pour i ← 1 à grandeur(V_grille1) faire
3   pour j ← 1 à grandeur(V_grille2) faire
4     Calcul du potentiel effectif au niveau du 2DEG;
5     V ← calcul_potentiel(paramètres_grilles, V_grille1, V_grille2) + V_oxyde;
6
7     Calcul de la densité électronique;
8     N_électrons ←
9       calcul_densité(interactions_poids, min_conduction, maille_dispo, dx_maille, V, info_int);
10
11   Assigner le nombre total d'électrons à occupation_élec;
12   occupation_élec(i, j) ← somme(N_électrons);
13   fin
14 fin
  
```

3.5 Stratégies d'accélération pour la simulation d'un diagramme de stabilité

Les algorithmes présentés aux sous-sections précédentes, si implémentés simplement et sans considérations particulières, font le calcul d'une occupation électronique en environ une seconde (implémentés sous Matlab [31][32] et exécutés sur un portable Dell XPS core i5 de neuf ans d'âge [33]). Cependant, les images désirées sont de haute résolution, exigeant de résoudre plus d'un million d'occupations par diagramme de stabilité. L'utilisation d'un modèle simple permet toutefois d'utiliser différentes stratégies d'accélération, présentées aux sous-sections suivantes, et d'ainsi tirer un réel avantage par rapport à un modèle plus précis mais tout aussi lent.

3.5.1 Critères de convergence de la densité électronique

Un des calculs les plus récurrents est le calcul par convergence de la densité électronique, étant répété pour chaque occupation. Le choix des critères pouvant déterminer si la convergence est atteinte est donc très important pour ne pas calculer une précision inutile, tout en s'assurant qu'elle soit suffisante pour la validité de la simulation.

La stratégie choisie est de ne pas utiliser de critères de convergence sur la densité

électronique elle-même, mais plutôt sur les îlots qu'elle crée. En effet, tel que montré à l'algorithme 1 ligne 6, la convergence est atteinte lorsque le nombre d'électrons et la position de chaque îlot sont constants. Cela permet de converger rapidement vers un nombre discret d'électrons, tout en s'affranchissant du fait que le modèle utilise une densité électronique continue. De plus, ces critères permettent de ne prendre en compte que les variables d'intérêt, soient les bornes et le nombre d'électrons des boîtes quantiques, lors de la décision de convergence. Finalement, les derniers avantages sont que la complexité de la convergence et la précision des calculs ne sont pas affectées par la précision de maille demandée, étant reliées directement à l'occupation électronique voulue sur le diagramme de stabilité.

3.5.2 Pré-calcul de l'intégrale de Fermi-Dirac

Le nombre de calculs nécessaires pour la convergence de la densité électronique étant optimisé, il est maintenant nécessaire de diminuer le temps de calcul d'un profil de densité. En effet, non seulement l'équation 3.2 est complexe³, mais son calcul est exécuté pour chaque élément de la maille, autant de fois qu'il en faut pour converger, et ce pour chaque calcul d'occupation nécessaire à la simulation d'un diagramme de stabilité. Une table de valeurs, contenant les résultats du calcul de la fonction $\eta(\epsilon'_0)$, est donc calculée à l'avance, sur une plage et une précision de ϵ'_0 jugées suffisantes. Ce pré-calcul a deux avantages considérables sur l'accélération des simulations :

- D'abord, une fois que le calcul de la fonction est fait et enregistré, le calcul de la densité électronique se réduit à un problème de recherche dans une liste. En effet, il suffit de chercher à quelle position de la liste se trouve la valeur de ϵ'_0 désirée, et de prendre la valeur de densité à cette position.
- Ensuite, le fait de connaître la plage ϵ'_0 sur laquelle est calculée la densité électronique permet de calculer directement la position d'une valeur de ϵ'_0 dans la liste grâce à l'équation 3.5. Ce dernier avantage permet de n'enregistrer que la première et de la dernière valeur de ϵ'_0 , ainsi que la liste de $\eta(\epsilon'_0)$ précalculée.

$$position = \frac{(\epsilon'_0 - \text{minimum}(plage(\epsilon'_0)) \cdot (\text{longueur}(plage(\epsilon'_0)) - 1))}{\text{maximum}(plage(\epsilon'_0)) - \text{minimum}(plage(\epsilon'_0))} \quad (3.5)$$

En conclusion, le calcul de l'intégrale de Fermi-Dirac dans l'espace est réduit à un calcul analytique d'entiers et à pointer dans une liste, et peut être fait simultanément pour chaque élément fini du dispositif simulé.

3. Il serait également possible de faire une résolution analytique de l'équation 3.2, étant donné la densité d'état constante. Cette méthode s'avère cependant plus longue pour les tests effectués, nécessitant une plus grande précision sur chacun des paramètres, et ce, à chaque itération.

3.5.3 Pré-calcul des poids d'interactions électroniques

Le prochain calcul exigeant le plus de temps de calcul est celui des interactions électroniques, nécessitant d'abord le calcul de l'équation 3.4, puis celui de l'intégrale $\int K(x, x')\eta(x')dx'$ de l'équation 3.3, et ce pour chaque élément fini du dispositif, et chaque itération de convergence.

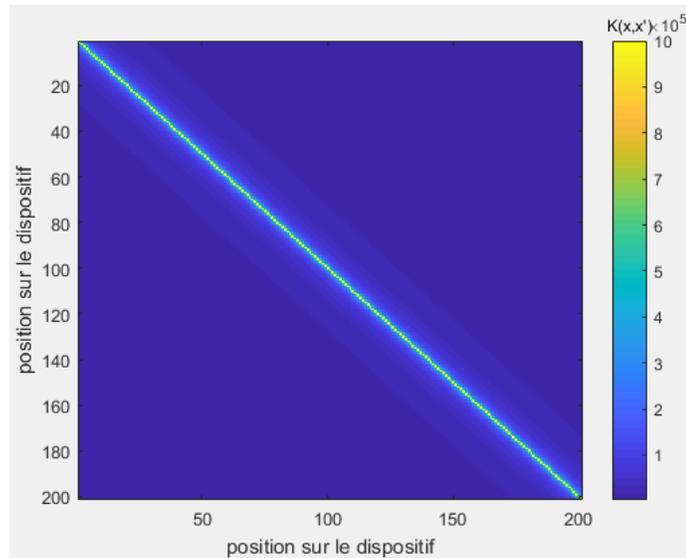


FIGURE 3.2 Exemple d'une matrice de $K(x, x')$ calculée à l'avance selon l'équation 3.4, avec 201 éléments finis, $K_0 = 5e - 3$ eV, et $\sigma = 5e - 9$ m

Pour limiter le nombre de calculs répétitifs, une matrice, représentant l'interaction $K(x, x')$ pour chaque élément fini, est calculée à l'avance (schématisée à la figure 3.2, et nommé *interactions_poids* dans les algorithmes). L'intégrale peut donc se calculer grâce au produit scalaire montré à l'équation 3.6 (ligne 8 de l'algorithme 1).

$$\text{interactions}_{\text{elec}} = \sum (\text{interactions}_{\text{poids}} \cdot \eta(x)) \quad (3.6)$$

3.5.4 Calcul partiel du diagramme de stabilité

À ce point, l'algorithme calculant l'occupation électronique pour des potentiels de grilles donnés est considéré optimisé. Le dernier noeud du problème est que le calcul d'un diagramme de stabilité complet nécessite le calcul d'innombrables occupations électroniques, souvent récurrentes, qui pourrait être facilement trouvées sans calcul. Il sera donc possible, pour un diagramme de stabilité d'une simple boîte quantique, d'utiliser une dernière stratégie diminuant jusqu'à cent fois le temps de simulation.

À la fin de cette dernière, l'information fournie par un diagramme se résume aux combinaisons de potentiels à appliquer sur les grilles de potentiel pour modifier l'occupation électronique dans le dispositif. Un calcul efficace serait donc de ne calculer que les combinaisons de potentiels modifiant l'occupation pour les plages de potentiels données, et d'interpoler le reste du diagramme grâce aux occupations calculées.

Comme montré jusqu'ici, les simulations restent classiques et idéales. De plus, il reste impossible d'augmenter le potentiel sur une grille et d'éjecter des électrons occupant déjà dans un îlot. Ces caractéristiques du modèle permettent de conclure que toutes les lignes de transitions électroniques :

- sont continues ;
- débutent sur le côté gauche ou le côté supérieur du diagramme ;
- ont un angle entre $-\pi/2$ et 0.

Il est donc possible, en calculant préalablement les occupations du côté gauche et supérieur d'un diagramme de stabilité, de suivre chaque transition jusqu'à son arrivée au côté inférieur (ou droit). Un exemple serait une transition allant de $N \rightarrow N + 1$ électrons dans le diagramme de stabilité. Le point de départ est la combinaison de potentiels donnant une occupation de N électrons, le plus à gauche possible dans le diagramme. Il faut ensuite suivre les deux étapes suivantes jusqu'à la fin de la transition, et pour chaque transition notée dans le calcul des côtés gauche et supérieur.

- Si l'occupation vaut N , il faut augmenter le potentiel en se déplaçant à droite dans le diagramme, jusqu'à retrouver $N + 1$;
- Si elle vaut $N + 1$, il faut diminuer le potentiel en se déplaçant en bas dans le diagramme, jusqu'à retrouver N .

Pour ce qui est du reste du diagramme (tout pixel n'étant pas de part et d'autre d'une transition), il peut être interpolé grâce aux transitions précédemment calculées. En effet, puisqu'il est impossible de diminuer le nombre d'électrons en augmentant le voltage, toute occupation entre deux transitions est donnée par la valeur de la transition la plus petite.

3.6 Liste des paramètres d'entrées du modèle physique

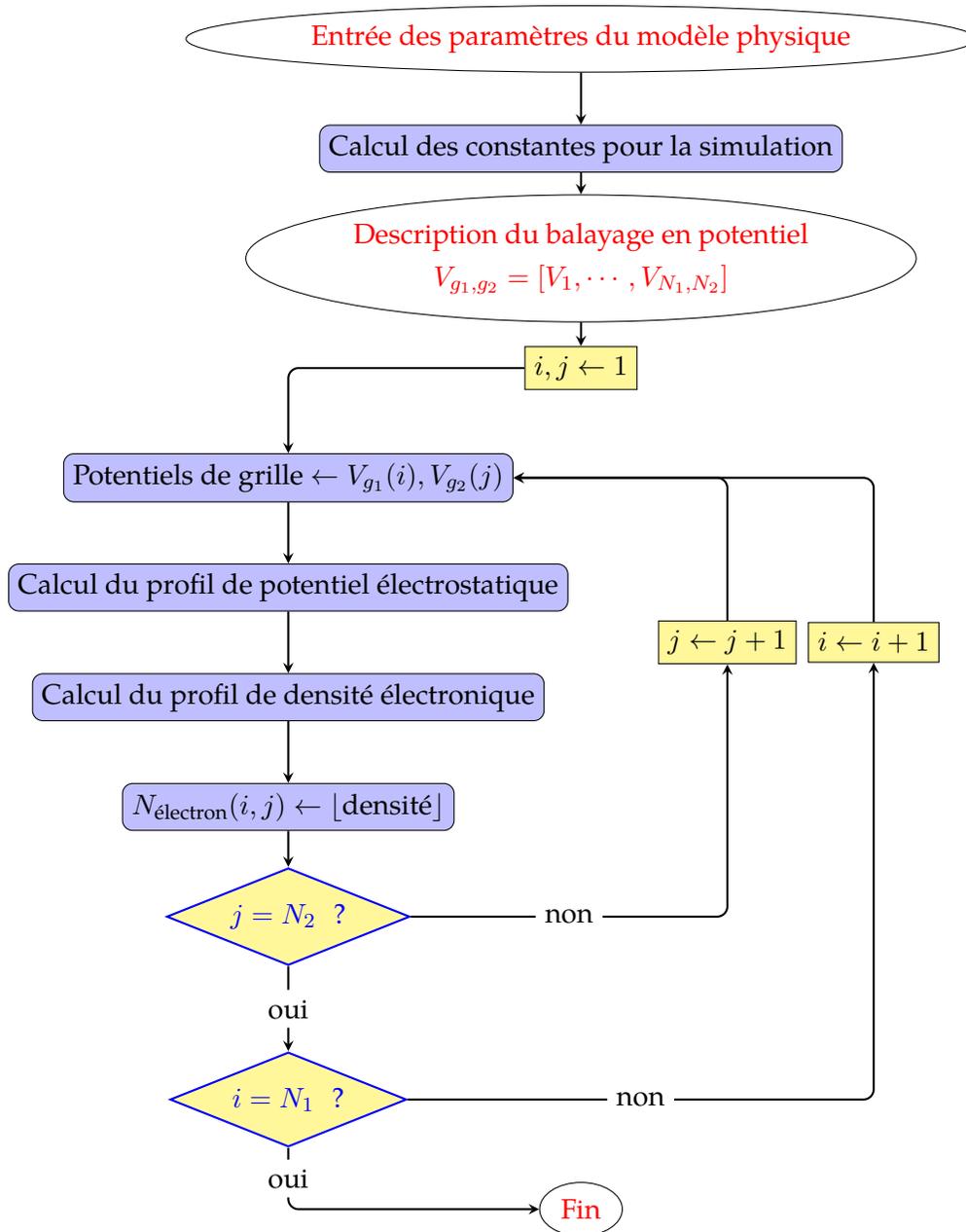
Paramètre	Définition	Exemple de valeur(s)
<i>Maillage</i>		
maille_dispo	Vecteur linéaire de positions, représentant les éléments finis du dispositif	$[-10^{-9}, \dots, 10^{-9}]$ [m]
<i>Constantes physique</i>		
g_0	Densité d'états (hypothèse : 2DEG)	1 [eV·nm ⁻¹] (1D)
k	Constante de Boltzmann	8.618e-5 [eV/K]
T	Température au niveau des électrons	0.1 [K]
E_f	Énergie de Fermi	0.1 [eV]
ϵ_0	Minimum de bande de conduction intrinsèque	0.01 [eV]
η_{oxyde}	Densité de charges présentes entre les grilles et les électrons	1e-9 [m ⁻²]
<i>Interaction entre les charges *</i>		
K_0	Potentiel d'interaction électronique à une distance σ des charges	5e3 [eV]
σ	Paramètre empêchant la divergence de l'équation 3.4 en $x = x'$	5e-9 [m]

TABLE 3.1 Liste des paramètres d'entrée relatifs au dispositif pour le calcul d'occupations électroniques et de diagrammes de stabilité.

*Les paramètres K_0 et σ ont simplement été choisis, pour l'instant, de façon à ne pas faire diverger le modèle, et valider par inspection visuel sur le diagramme de stabilité simulé. De façon conceptuelle, K_0 représente l'amplitude maximale du potentiel d'interaction qu'une charge de 1 eV peut appliquer en sa position. Le paramètre σ permet ensuite de contrôler si le profil de potentiel engendré par cette charge sera plus évasé ou plus effilé. Intuitivement, il est donc possible de comprendre qu'un large ensemble de combinaisons de K_0 et σ peut satisfaire la convergence du modèle, en restant tout autant physique. Dans le cas présent, la mesure désirée ne se base que sur l'image, d'où la validation visuelle.

Paramètre	Définition	Exemple de valeur(s)
<i>Construction d'une grille</i>		
V_grille	Potentiel appliqué sur la grille de potentiel	0.5 [V]
x ₀	Position de la grille sur le dispositif	-500e-9 [m]
R _g	Rayon de la grille	5e-9 [m]
l	Longueur de la grille	5e-6 [m]
hauteur_grilles	Distance entre la grille et les élec.	50e-9 [m]
cond_inf	Distance où le potentiel généré par une grille est considérée nul	2000e-9 [m]
<i>Structure de paramètres utilisée dans l'algorithme</i>		
paramètres_grilles	Structure contenant les six constantes ci-dessus et la fonction de potentiel $f(r)$ de l'équation 3.1. pour chaque grille	-
<i>Information de balayage de potentiels</i>		
V_grille1 (et 2)	Vecteur contenant toutes les valeurs à tester pour la grille 1 (et 2)	[0, ..., 0.1] [V]
n1 (et 2)	Position de la grille 1 (et 2) à balayer dans la structure <i>paramètres_grilles</i> .	2 [-]
CC	Matrice de capacités-croisées entre les grilles et les boîtes quantiques permettant l'utilisation de grilles virtuelles. Une matrice identité [n_grille par n_grille] utilisera les grilles réelles.	I

TABLE 3.2 Liste des paramètres d'entrée relatifs aux grilles pour le calcul d'occupations électroniques et de diagrammes de stabilité.



Organigramme des étapes de calcul d'un diagramme de stabilité simulé grâce au modèle Thomas-Fermi

Chapitre 4

Augmentation de données par techniques de traitement d'images

L'augmentation de données [34] est le terme général référant au pré-traitement d'une base de données, visant à augmenter la quantité d'informations qu'il sera possible d'extraire de celle-ci. Ce pré-traitement peut consister, par exemple, en une série d'opérations mathématiques, permettant d'augmenter la richesse de l'information à traiter. Il peut également s'agir de doubler la taille d'une base de données d'images, en copiant chacune d'elles et en lui faisant subir une rotation de 180 degrés¹.

En ce qui concerne les diagrammes de stabilité, ce seront des techniques de traitement d'images qui seront utilisées à titre d'augmentation de données, afin d'augmenter le réalisme des simulations et de faciliter la génération de données supplémentaires. En effet, bien que physique, le modèle choisi reste presque uniquement classique, et ne produit pas de bruit, d'imperfections, d'impacts de pré-traitement de mesure, ou autres types de défauts pouvant affecter un diagramme de stabilité expérimental. L'utilisation du simulateur pour entraîner un algorithme d'apprentissage performant sur des données expérimentales devient donc risqué, voire impossible. Afin d'y arriver, des techniques de traitement d'images ont été implémentées pour ajouter des caractéristiques qui ne sont pas prises en compte par le modèle. La stratégie est donc simple : étant donné que la réalité est longue et difficile à calculer, mais qu'elle est connue et facile à interpréter, il suffit de générer manuellement tous les artefacts réels qu'y ne sont pas compris dans le modèle.

1. Cette stratégie est couramment utilisée en imagerie médicale ou satellitaire, et de plusieurs angles, afin d'augmenter la taille de la base de données. Cela permet, par exemple, d'avoir des photos d'un lac de tous les angles voulus à partir d'une seule photo.

À noter qu'à ce moment, un choix important est fait pour l'utilisation du simulateur. En effet, une grande quantité d'artefacts seront présents sur un résultat final, donnant un bon rendu visuel. Cependant, ces artefacts ne pourront pas être reliés à des grandeurs physiques. Bien que limitatif, ce choix apporte une grande simplicité au problème, dont il est possible de tirer avantage. D'abord, il devient possible de créer une grande variété de diagrammes de stabilité, avec plusieurs effets réels, tout en s'affranchissant du long temps de calcul qu'une simulation générant elle-même ces artefacts prendrait. Ensuite, si chaque fonction de traitement d'images est indépendante, il est possible de contrôler la proportion et les caractéristiques de chaque artefact, en vue de la génération d'une grande quantité de diagrammes de stabilité. Finalement, les fonctions indépendantes permettent d'utiliser le simulateur pour tous types de mesures ou de dispositifs, n'ayant qu'à ajouter ou à supprimer certaines fonctions s'il y a un changement expérimental important.

Le simulateur devient ainsi un outil intuitif, permettant à tout physicien de dessiner «à la main» des lignes sur un diagramme. Les possibilités de retrouver des lignes à un certain angle, d'être espacées d'une certaine distance, ou autres caractéristiques géométriques, sont toutefois restreintes par des équations physiques. Il est à noter que l'idée aurait pu être poussée à l'extrême, en générant des diagrammes de stabilité simplement en dessinant des lignes d'angles et d'espacements arbitraires grâce, par exemple, à un logiciel de dessin. Cependant, à ce moment, certains diagrammes de stabilité générés seront complètement différents de ce qui peut être simulé par le modèle de Thomas-Fermi, ou mesuré expérimentalement. Ceci implique que l'algorithme d'apprentissage devra apprendre à reconnaître des diagrammes de stabilités semblant être réels, ainsi que d'autres ne l'étant pas du tout. Il s'agit donc intuitivement d'un problème plus complexe à résoudre, et nécessiterait possiblement davantage de données d'entraînement pour obtenir des résultats concluant. C'est pourquoi, dans le travail présent, la preuve de concept se limite à la génération de diagrammes par un modèle physique, suivie de fonctions de traitement d'images, lesquelles sont détaillées dans les prochaines sous-sections.

4.1 Passage de l'occupation électronique à la mesure expérimentale

Évidemment, en laboratoire, ce n'est pas l'occupation électronique qui est mesurée mais plutôt le signal du transistor mono-électronique utilisé en détection de charges. Les données sont ensuite traitées par un algorithme implémenté par Maxime Lapointe-Major, résultant en une image binaire, où les points du diagramme de stabilité donnant lieu à une transition électronique sont identifiés par un « 1 », alors que les autres sont identifiés par un « 0 ».

En d'autres mots, un pixel vaut « 1 » si, pour la combinaison de potentiels lui étant associée, une modification de l'occupation électronique du dispositif a lieu. Sinon, une valeur de « 0 » lui est attribuée. Dans le cas des simulations, c'est l'occupation totale du dispositif qui est notée pour chaque combinaison de potentiels. Ainsi, une dérivée numérique (voir algorithme 5) permet de donner la valeur de « 1 » à tout pixel où une transition a lieu, et de mettre les autres égal à « 0 », l'occupation étant constante entre deux transitions. Un exemple du résultat de cette transformation est donné à la figure 4.1.

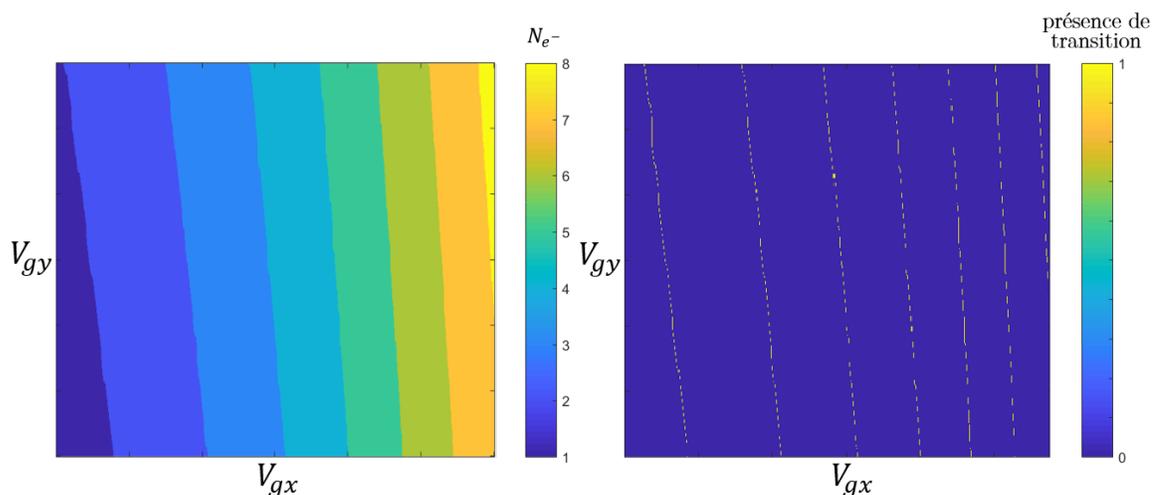


FIGURE 4.1 Exemple d'une occupation électronique transformée en mesure binaire. À noter que les lignes sur l'image de droite sont supposées continues, et que les brisures ici présentes sont des artefacts visuels dues à l'extraction de la figure.

Le diagramme ainsi modifié est considéré comme étant le signal mesuré en détection de charges, suivi du traitement de l'algorithme de Lapointe-Major. Il est à noter que, suite à ce premier traitement d'images, le simulateur est fixé dans un régime où toutes transitions électroniques dans la plage de potentiels explorée sont visibles. Ce n'est évidemment pas le cas en laboratoire, imposant une première limite expérimentale : toutes les transitions doivent être observables.

Algorithme 5 : Dérivée de l'occupation électronique

Entrées : occupation_élec

Sorties : signal

- 1 Calcul des dérivées de chaque dimension ;
 - 2 $d_occupation_i \leftarrow diff(occupation_élec, dimension \leftarrow i)$;
 - 3 $d_occupation_j \leftarrow diff(occ, dimension \leftarrow j)$;
 - 4 Calcul du signal mesuré en détection de charges ;
 - 5 $signal \leftarrow d_occupation_i + d_occupation_j$;
 - 6 Rendre binaire le signal calculé ;
 - 7 $signal(signal \geq 1) = 1$
-

4.2 Ajouts de défauts physiques dans le dispositif

Jusqu'à présent, un dispositif idéal a été simulé, fournissant l'occupation électronique des boîtes quantiques ainsi créées. Dans la réalité, les matériaux ne sont pas complètement purs, et peuvent comporter des défauts près des boîtes, modifiant leur environnement électrostatique. Ces défauts sont, par exemple, des atomes manquants dans le réseau cristallin, ou des atomes étrangers au matériau utilisé. En modifiant les potentiels de grilles, il peut arriver que des charges se logent en ces endroits, causant des artefacts sur un diagramme de stabilité expérimental. Trois cas de figure sont illustrés ci-après, ayant chacun un impact différent sur la mesure.

4.2.1 Un défaut couplé au détecteur de charges, mais non à la boîte quantique

Lorsqu'une charge est captée par le défaut, le changement de potentiel qu'elle engendre est perçu par le détecteur. Ce changement agit comme une boîte quantique qui capte un électron, et crée une ligne de transition dans le diagramme de stabilité (des exemples expérimentaux sont montrés à la figure 4.2. L'angle de cette transition est donné par le couplage entre les grilles balayées et le défaut. La position du défaut étant aléatoire dans un dispositif, l'angle de la ligne qu'il génère est arbitraire. L'algorithme permettant de reproduire cet effet trace donc simplement une ligne dans un diagramme (algorithme 6), paramétrée par un angle et une ordonnée à l'origine.

Algorithme 6 : Ajout d'une ligne sur le signal

Entrées : signal, hauteur_début_défaut, angle

Sorties : signal_modifié

```

1 Calcul de l'emplacement du premier pixel pour le début du traçage;
2  $i0 \leftarrow \min(\text{taille}(\text{signal}) \text{ ou } \text{taille}(\text{signal}) \cdot \text{hauteur\_début\_défaut});$ 
3  $j0 \leftarrow \max(1 \text{ ou } (1 - \text{taille}(\text{signal}) \cdot \text{hauteur\_début\_défaut}) \cdot \tan(\text{angle}));$ 

4 Calcul de la perte en hauteur pour l'avancement d'un pixel en j;
5 compteur_pente  $\leftarrow |1/\tan(\text{angle})|;$ 
6 pente  $\leftarrow \text{compteur\_pente} - 1$ 

7 Initialisation du signal modifié, des compteurs, et traçage de la ligne;
8 signal_modifié  $\leftarrow \text{signal};$ 
9  $i \leftarrow i0;$ 
10  $j \leftarrow j0;$ 
11 tant que  $i > 0 \ \& \ j < \text{taille}(\text{signal})$  faire
12   | signal_modifié( $i, j$ )  $\leftarrow 1;$ 
13   | si pente  $> 0$  alors
14     |    $j \leftarrow j + 1;$ 
15     |   pente  $\leftarrow \text{pente} - 1;$ 
16   | sinon
17     |    $i \leftarrow i - 1;$ 
18     |   pente  $\leftarrow \text{pente} + \text{compteur\_pente};$ 
19   | fin
20 fin

```

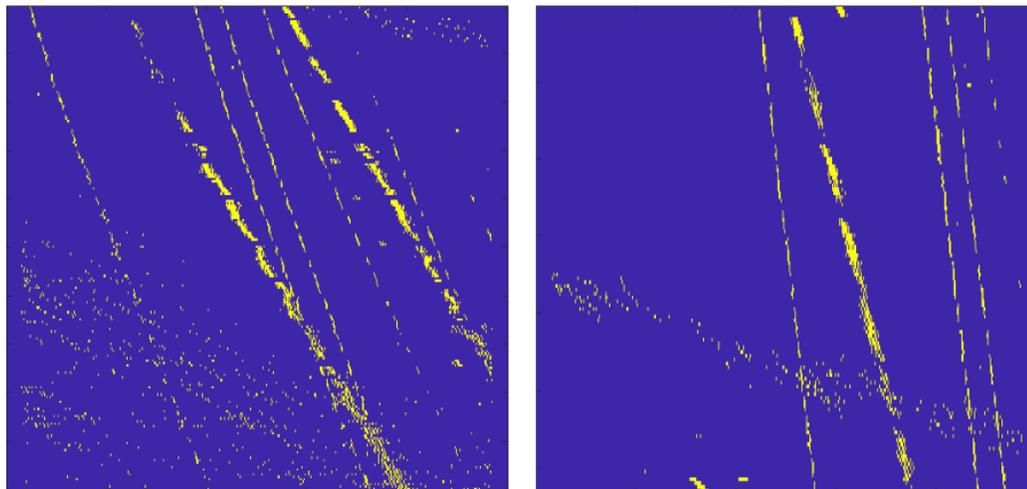


FIGURE 4.2 Exemple d'un diagramme de stabilité expérimental où une charge parasite s'est couplée au détecteur de charge pendant la mesure.

4.2.2 Un défaut couplé à la boîte quantique, mais non au détecteur de charges

À la différence du dernier cas de figure, le défaut ici considéré n'est pas couplé au détecteur de charges. Ainsi, aucune ligne parasite n'apparaît sur le diagramme de stabilité. Cependant, étant couplées à la boîte quantique, les charges captées modifient le potentiel effectif de cette dernière. Afin de lutter contre ce potentiel et d'ajouter un électron dans la boîte, une énergie supplémentaire doit être fournie par les grilles de potentiel, faisant en sorte qu'une ligne de transition puisse être observée plus loin que prévu sur le diagramme de stabilité. Les lignes de transition subissent donc une translation pour une combinaison de potentiels captant une charge dans une impureté, comme illustré à la figure 4.3.

De façon similaire à l'algorithme précédent, celui-ci considère en entrée les paramètres d'angle et d'ordonnée à l'origine de l'artefact à créer, en plus d'une proportion de l'abscisse comme déplacement des pixels pour la translation. Ce nombre représente l'intensité du couplage : plus une impureté est près de la boîte, plus son couplage est important, et plus la translation sera importante. La modification est ensuite faite suivant la ligne du défaut et le nombre de pixels donnés (voir algorithme 7). Un exemple de résultat est montré à la figure 4.4.

Algorithme 7 : Translation du diagramme de stabilité

Entrées : signal, hauteur_début_défaut, angle, déplacement

Sorties : signal_modifié

```

1 Calcul du déplacement engendré par la translation ;
2 déplacement ← prop_déplacement_translation · taille(signal) ;

3 Calcul de l'emplacement du premier pixel pour le début de la translation ;
4  $j_0 \leftarrow \min(\text{taille}(\text{signal}) \text{ ou } \text{hauteur\_début\_défaut} \cdot \text{taille}(\text{signal}) \cdot \tan(\text{angle}))$  ;
5  $i_0 \leftarrow \max(1 \text{ ou } \text{taille}(\text{signal}) \cdot \tan(\text{angle}))$  ;

6 Calcul de l'emplacement du dernier pixel pour la fin de la translation ;
7  $j_{fin} \leftarrow \min(1 \text{ ou } j_0 + \text{taille}(\text{signal}) \cdot \tan(\text{angle}))$  ;

8 Calcul du gain en hauteur pour le recul d'un pixel en j ;
9 élément_hauteur_translation ←  $(i_0 + 1) / (j_{fin} - j_0)$  ;
10 compteur_hauteur_translation ← élément_hauteur_translation ;

11 Initialisation du signal modifié et translation du diagramme ;
12 signal_modifié ← signal ;
13 pour  $j \leftarrow j_0$  à déplacement par bond de -1 faire
14   tant que compteur_hauteur_translation < 0 faire
15     compteur_hauteur_translation ← compteur_hauteur_translation + ;
16      $i_0 \leftarrow i_0 + 1$  ;
17   fin
18   signal_modifié(1 à  $i_0, j$ ) ← signal(1 à  $i_0, j - \text{déplacement}$ ) ;
19   compteur_hauteur_translation ← compteur_hauteur_translation - 1 ;
20 fin

21 Calcul de la partie du diagramme manquante due à la translation ;
22  $dv \leftarrow \text{balayage\_potentiel\_initial\_j}(\text{déplacement} + 1) - \text{balayage\_potentiel\_initial\_j}(1)$  ;
23 balayage_potentiel_i ← balayage_potentiel_initial_i(1 à  $i_0$ ) ;
24 balayage_potentiel_j ← balayage_potentiel_initial_j(1 à déplacement) - dv ;
25 signal_modifié(1 à  $i_0, 1$  à déplacement) ← algorithme 4 et 5 ;

```

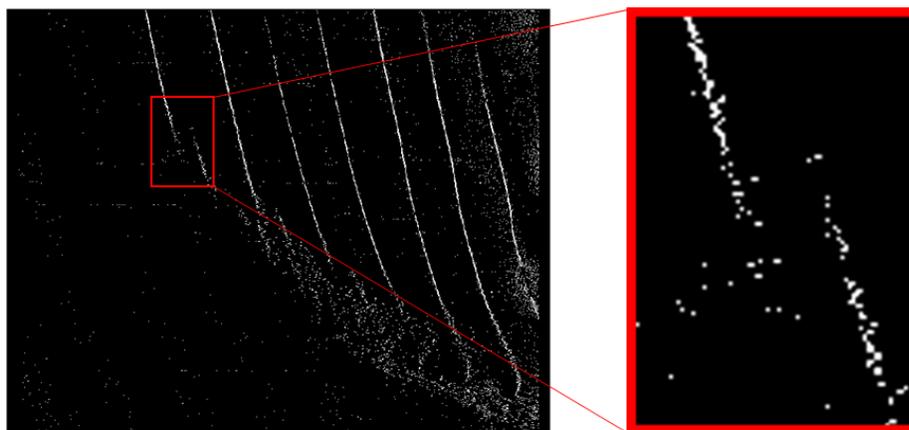


FIGURE 4.3 Exemple d'un diagramme de stabilité expérimental où une charge parasite s'est couplée à la boîte quantique pendant la mesure.

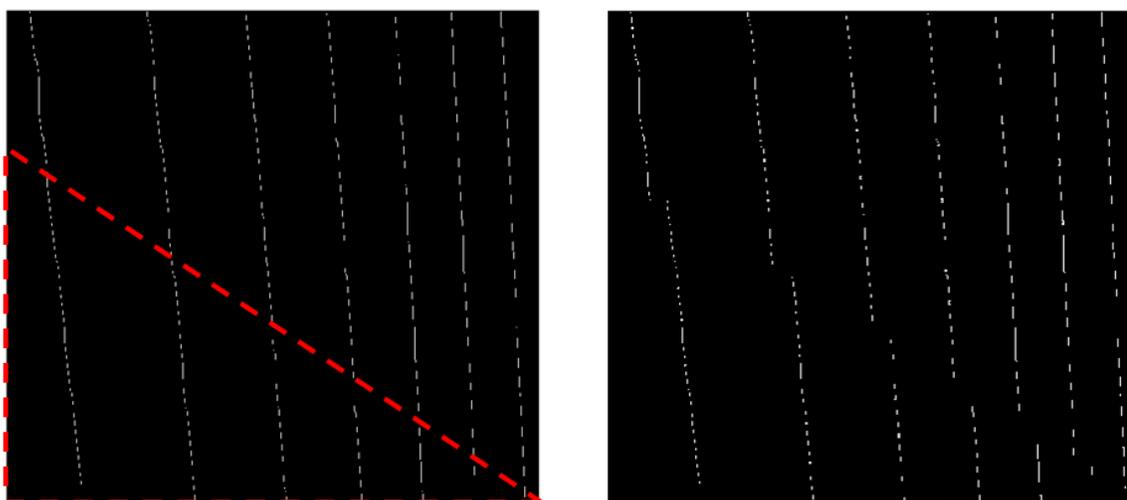


FIGURE 4.4 Exemple du résultat de la fonction de traitement d'images représentant un défaut couplé à la boîte quantique mais non au détecteur de charge. Le triangle en pointillé rouge sur le diagramme de gauche représente la translation à faire, et le résultat est affiché à droite.

4.2.3 Un défaut couplé au détecteur de charges et à la boîte quantique

Dans le cas présent, la charge piégée est assez près de la boîte quantique et du détecteur pour interagir avec les deux. Donc, en plus d'ajouter une ligne de signal parasite au diagramme, cette charge a deux impacts sur les lignes de transitions déjà présentes.

D'abord, chaque fois qu'une charge est captée par un piège, la ligne de transition de la boîte subit une translation. Cet effet est aussi vrai à l'inverse, causant une translation de la ligne de transition du défaut. Ensuite, les interactions étant régies par la mécanique

quantique, il peut y avoir présence d'anticroisements entre les deux lignes de transition. Dans ce cas, deux lignes qui se croisent avec la forme d'un "X" se transforment en deux hyperboles vis-à-vis (voir figure 4.5).

L'algorithme qui génère ce type d'artefact requiert deux parties séparées. D'abord, une ligne de défaut est ajoutée au diagramme, et les translations sont faites sur celle-ci ainsi que sur les lignes de transition, résultant en un premier anticroisement (algorithme 8). Dans le cas où une charge est très couplée à la boîte, l'anticroisement à dessiner est plus prononcé, exigeant de « dessiner » de meilleures hyperboles. Cette tâche est réalisée par une génération Steiner [35], avec l'algorithme 9. Des exemples du résultat de l'ajout d'une ligne et de ses anticroisements sont montrés à la 4.7.

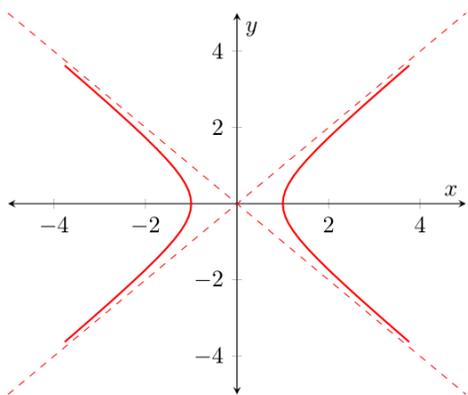


FIGURE 4.5 Exemple d'un «X» se transformant en deux hyperboles.

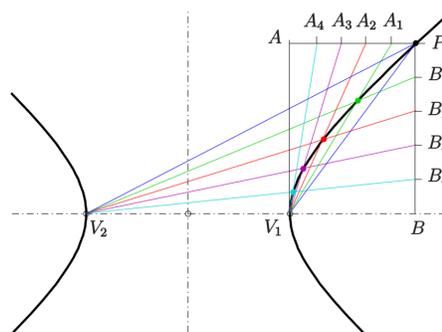


FIGURE 4.6 Utilisation de la génération d'hyperbole Steiner pour créer l'anticroisement.

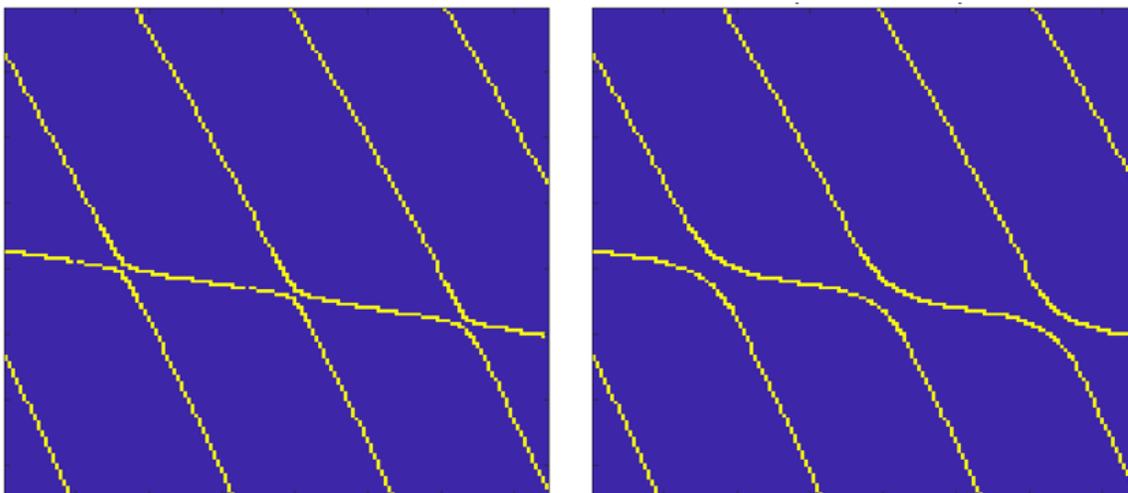


FIGURE 4.7 Exemple de résultats de la fonction de traitement d'images représentant un défaut couplé à la boîte quantique et au détecteur de charges.

Algorithme 8 : Ajout d'une ligne et de ses anticroisements sur le signal (Partie 1)

Entrées : signal, hauteur_début_défaut, angle_défaut

Sorties : signal_modifié

1 *Création de la ligne de défaut et de la carte de translation: voir algorithme 6, et ajouter les commandes suivantes aux lignes indiquées.;*

2 ligne 8 : `carte_translation ← zeros(taille(signal));`

3 ligne 12 : `carte_translation(1 à i , j) ← 1;`

4 ligne 19 : `si signal = 1 : $i ← i + d$, indices_croisements ← [$i - 2d/3$, $j + d/2$];`

5 Algorithme 6 ← signal_défaut, carte_translation;

6 *Déplacement horizontal des lignes de transition sur le diagramme de stabilité;*

7 **pour** $i ← 1$ à `taille(signal)` **faire**

8 **pour** $j ← 1$ à `taille(signal)` **faire**

9 **si** `carte_translation(i , j) = 1` **alors**

10 `signal_modifié(i , $j - d$) ← signal(i , j);`

11 **sinon**

12 `signal_modifié(i , j) ← signal(i , j);`

13 **fin**

14 **fin**

15 **fin**

16 *Translation du signal du défaut pour que la ligne s'ajuste à celles du diagramme de stabilité;*

17 `signal_défaut ← signal_défaut(d pixel à droite);`

18 *Création du nouveau diagramme de stabilité;*

19 `signal_modifié ← signal_modifié + signal_défaut;`

20 *Enlever les artefact créés au centre des croisements (fonction `bwmorph` de la bibliothèque de traitements d'image de Matlab [32]);*

21 `signal_modifié ← bwmorph(signal_modifié, 'spur', d);`

22 `signal_modifié ← bwmorph(signal_modifié, 'fill');`

23 `signal_modifié ← bwmorph(signal_modifié, 'skel');`

Algorithme 9 : Ajout d'une ligne et de ses anticroisements sur le signal (Partie 2)

Entrées : signal, d_attache

Sorties : signal_modifié

```

1 Stratégie :
  Utiliser la génération d'hyperbole Steiner (voir figure 4.6) pour
  créer une parcelle correctrice sur le diagramme de stabilité.

2 Initialisation du signal à modifier ;
3 signal_modifié ← signal ;

4 Remplacer les anticroisements précédents par des parcelles d'hyperboles générées par la
  génération Steiner ;
5 pour chaque indices_croisements notés à l'algorithme 8 faire
6   Initialisation des valeurs nécessaires au traçage de l'hyperbole (dans le référentiel donné
   de la figure 4.6);
7    $V_1, V_2 \leftarrow \text{indices\_croisements} \pm d/2$ ;
8    $P \leftarrow \text{pixel où signal} = 1 \text{ à } V_1 + d\_attache$ ;
9    $[A_1, \dots, A_N] \leftarrow \text{coordonnées } j \text{ de } [P \rightarrow V_1]$ ;
10   $[B_1, \dots, B_N] \leftarrow \text{coordonnées } i \text{ de } [P \rightarrow 0]$ ;

11  Création de la parcelle contenant l'hyperbole à tracer sur le diagramme ;
12  parcelle ← zeros(taille(signal));
13  pour  $i \leftarrow 1$  à  $N$  faire
14    si  $\overline{V_1 A_i}$  croise  $\overline{V_2 B_i}$  alors
15      | parcelle(croisement) ← 1;
16    fin
17  fin

18  Collage de l'hyperbole sur le diagramme de stabilité;
19  parcelle_à_coller ← rectangle d'aire minimale contenant tous les 1 de parcelle;
20  signal_modifié(parcelle_à_coller) ← parcelle_à_coller;
21 fin

```

4.3 Effets du taux tunnel réservoir - boîte quantique

Un autre effet physique non considéré par le simulateur est la façon dont les électrons se rendent dans la boîte quantique. En effet, bien qu'un puits de potentiel ait été créé, et qu'il soit disposé à accumuler un électron, ce dernier doit tout de même franchir la barrière de potentiel séparant le puits du réservoir où il se trouve. Le passage de cet électron est dicté par l'effet tunnel, qui est dépendant de la hauteur de cette barrière, et qui est à son tour dépendante des potentiels appliqués sur les grilles du dispositif. Puisqu'un diagramme de stabilité est mesuré grâce à un balayage en potentiel sur ces grilles, les propriétés du passage tunnel peuvent en être affectées, et ainsi créer des artefacts sur le signal mesuré.

4.3.1 Transmission de la barrière tunnel

Une première propriété de l'effet tunnel affectant les diagrammes de stabilité est son caractère probabiliste quant au passage d'un électron. En effet, la transmission (T) d'un électron au travers une barrière de potentiel est une fonction qui décroît exponentiellement avec la hauteur de la barrière. Puisque l'accumulation des électrons se fait par l'augmentation des potentiels de grilles, qui a pour effet de diminuer la barrière entre la boîte quantique et le réservoir, la transmission aura également tendance à augmenter.

Sur un diagramme de stabilité, cet effet se traduit par l'épaississement des lignes de transition avec l'augmentation des potentiels de grilles, étant donné une plus grande facilité à transmettre. Cet épaississement dépend également de l'intensité avec laquelle la variation du potentiel d'une grille fait varier la hauteur de la barrière. Une grille située loin de la barrière aura donc un moins grand impact sur l'épaisseur de la ligne qu'une grille située plus près. Ainsi, l'épaisseur des lignes, et donc la transmission, varie avec la somme des contributions exponentielles des grilles balayées (équation 4.1) [36].

$$T \propto \exp [AV_{g1} + BV_{g2}] \quad (4.1)$$

L'équation 4.1 est définie par les paramètres A et B , représentent les couplages entre les grilles et la barrière tunnel séparant la boîte quantique et le réservoir. Visuellement, cet effet s'observe par des lignes plus larges sur les parties supérieures et droites du diagramme de stabilité.

Les diagrammes de stabilité simulés ont, jusqu'à présent, des lignes de un pixel, provenant de la dérivée des occupations calculées. L'algorithme générant l'épaississement de ces

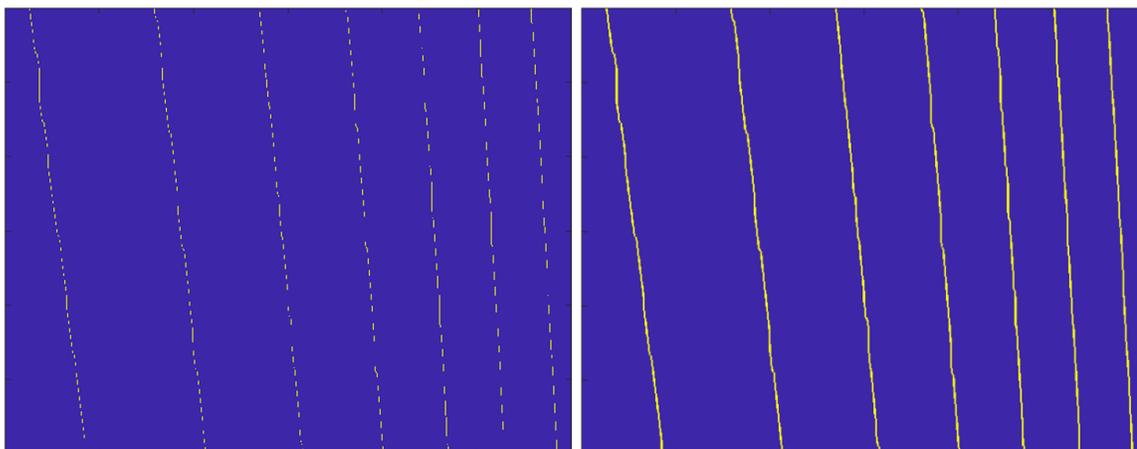


FIGURE 4.8 Exemple de l'élargissement des lignes d'un diagramme de stabilité. À noter que les lignes sur l'image de gauche sont supposées continues, et que les brisures ici présentes sont des artefacts visuels dues à l'extraction de la figure.

lignes prend en entrée le nombre de pixel minimum désiré pour la largeur d'une ligne, le nombre de pixel maximum désiré, les coefficients A et B , ainsi qu'un nombre de moyennage. Ce dernier paramètre permet de contrôler le niveau de bruit de l'élargissement de la ligne, en plus d'être à la base de la technique montrée à l'algorithme 25. Un exemple de cette modification est illustré à la figure 4.8.

Il est cependant évident qu'un simple élargissement des lignes de transition fait de la sorte ne peut pas reproduire l'entièreté des caractéristiques réelles. En effet, le bruit ajouté en fonction du moyennage ne modifie que les bordures des lignes de transition, et n'affecte aucun pixel interne. Pour permettre la modification d'une ligne entière, la même carte de taux tunnel est utilisée, ainsi qu'une carte de valeurs aléatoires, afin d'effacer un certain nombre de pixels, ayant le même comportement que celui de l'élargissement. Le diagramme ainsi modifié comportera des lignes de transition de différentes largeurs, ainsi qu'une diminution du signal mesuré, suivant le comportement exponentiel du taux tunnel avec la modification des potentiels de grilles.

4.3.2 Taux tunnel vs fréquence des mesures

Un seconde propriété de l'effet tunnel ayant également un impact sur le diagramme de stabilité est le taux auquel les électrons peuvent franchir la barrière, appelé *taux tunnel*. Il augmente exponentiellement avec la réduction de la barrière tunnel, diminuant de la même manière le temps d'attente moyen avant qu'un électron se loge dans la boîte quantique. Un problème survient à l'inverse, lorsque la hauteur de la barrière devient si grande que le taux

Algorithme 10 : Élargissement des lignes de transition (Explications)

Entrées : signal, largeur_max, largeur_min, A, B, n_moyennage

Sorties : signal_modifié

1 **But** :

Avoir des lignes de transition d'une largeur de largeur_max à l'endroit où la transmission est maximale, et d'une largeur de largeur_min à l'endroit où elle est minimale.

2 **Description** :

Deux matrices de la taille de celle du signal sont créées. La première représente la probabilité de transmission d'un pixel étant donné les potentiels appliqués, et est appelée carte_taux_tunnel. La seconde donne un chiffre aléatoire entre 0 et 1 à chaque pixel (transmission_élec). La matrice de transmission s'écrit comme suit :

$$\text{carte_taux_tunnel} = \begin{pmatrix} & & \dots & 1 \\ & \vdots & \ddots & \vdots \\ \frac{\text{largeur_min}}{\text{largeur_max}} & & \dots & \end{pmatrix}$$

La probabilité d'élargissement est donc de $p_{min} = \left[\frac{\text{largeur_min}}{\text{largeur_max}} \right]$ pour le coin inférieur gauche, et de 1 pour le coin supérieur droit. Il y aura élargissement si $\text{transmission_élec} < \text{carte_taux_tunnel}$, puis l'algorithme est répété « largeur_max » fois : la ligne du coin droit aura une largeur de $\text{largeur_max} \cdot 1 = \text{largeur_max}$, et celle du coin gauche aura une largeur de $\text{largeur_max} \cdot p_{min} = \text{largeur_min}$.

Algorithme 10 : Élargissement des lignes de transition

Entrées : signal, largeur_max, largeur_min, A, B, n_moyennage

Sorties : signal_modifié

```

1 Initialisation de la matrice de signal modifié;
2 signal_modifié ← signal

3 Calcul de l'élargissement des lignes à faire;
4 pour i ← 1 à n_moyennage faire
5   pour j ← 1 à largeur_max faire
6     Création d'une « carte » de taux tunnel suivant l'équation 4.1 ;
7     impact_grille_i ← A · linspace(-1,0), impact_grille_j ← B · linspace(-1,0);
8     carte_taux_tunnel ← normaliser(exp(impact_grille_i + impact_grille_j));

9     Calculer la probabilité d'élargissement de la carte de taux tunnel;
10    carte_taux_tunnel ← (1 - largeur_min/largeur_max) · carte_taux_tunnel;
11    carte_taux_tunnel ← carte_taux_tunnel + 1 - max(carte_taux_tunnel);

12    Création de la carte aléatoire pour transmettre les électrons;
13    transmission_élec ← matrice de la taille de carte_taux_tunnel, d'entiers
        aléatoires entre 0 et 1;

14    Détermination des pixels qui élargiront la ligne;
15    avancement_pixel ← transmission_élec < carte_taux_tunnel & signal == 1;

16    Élargir la ligne si la transmission de l'électron est plus grande que celle du taux
        tunnel;
17    si j mod 2 == 1 alors
18      signal_modifié ← signal_modifié + avancement_pixel(indice + 1); // Vers
        la droite;
19    sinon
20      signal_modifié ← signal_modifié(indice + 1) + avancement_pixel; // Vers
        la gauche;
21    fin
22  fin
23 fin

24 Rendre le signal binaire en fonction du nombre de moyennage fait;
25 signal_modifié(signal_modifié ≥ 0.95 · n_moyennage) ← 1, les autres sont mis à 0;

```

tunnel se compare à la fréquence de la mesure.

En effet, mesurer un diagramme de stabilité est une longue tâche, et le temps entre deux mesures est déterminé, entre autres, afin d'obtenir un temps total raisonnable. Il peut donc arriver, au cours d'une mesure, d'avoir des taux tunnel de l'ordre de la fréquence à laquelle est répétée la mesure. À ce moment, même si la boîte quantique a la possibilité d'accueillir un électron supplémentaire pour la combinaison de potentiels appliqués, un signal ne sera pas nécessairement mesuré immédiatement. Il pourra toutefois être mesuré plus tard, lorsqu'un électron aura finalement traversé la barrière, ou même jamais, si le temps que l'évènement se produise est plus grand que le temps de mesure du diagramme. À noter que le temps avant de pouvoir mesurer le signal est lui aussi statistique, étant estimé comme une gaussienne centrée sur la période tunnel ($1/\text{taux tunnel}$).

Sur le diagramme de stabilité, il est possible d'observer une (ou plusieurs) courbure des lignes de transition. Une mesure trop rapide, étant l'une des causes de ces courbures, donne un caractère exponentiel décroissant aux lignes de transition, ainsi qu'un évasement de ces dernières (figure 4.9). En effet, à faibles potentiels de grilles, la hauteur de la barrière est grande, et provoque un retard sur la mesure. Si cette mesure est faite de la gauche vers la droite, le signal sera mesuré à droite de ce qui était attendu. En augmentant le potentiel (se déplaçant vers le haut dans le diagramme de stabilité), la barrière diminue en hauteur (augmentant le taux tunnel), diminuant ainsi le retard de mesure. Cet effet se produit jusqu'à ce que le taux tunnel devienne beaucoup plus grand que la fréquence de mesure, n'ayant plus d'impact sur le diagramme. De plus, en considérant le fait que le temps nécessaire pour mesurer l'électron est donné par une distribution gaussienne, le déplacement d'un pixel de signal devient lui aussi gaussien. Il peut donc se créer un évasement des lignes de transition, suivant la moyenne et l'écart type de la distribution. Il est à noter qu'un signal ne peut être mesuré avant que les potentiels n'aient été appliqués, et donc qu'aucun déplacement vers la gauche n'est possible dans l'exemple de la mesure gauche \rightarrow droite donnée. L'algorithme 11 montre comment réaliser ce genre de modification, toujours dans le cas d'une mesure gauche \rightarrow droite.

Algorithme 11 : Retard du signal de l'électron pour des mesures à fréquence trop rapide

Entrées : signal, A, B, prop_déplacement_moy, prop_écart-type
Sorties : signal_modifié

- 1 Paramètres de la fonction normale;
- 2 déplacement_moyen \leftarrow taille(signal) · prop_déplacement_moy;
- 3 écart-type \leftarrow taille(signal) · prop_écart-type;
- 4 Création d'une « carte » de taux tunnel suivant l'équation 4.1 ;
- 5 impact_grille_i \leftarrow A · linspace(0, -1), impact_grille_j \leftarrow B · linspace(0, -1);
- 6 carte_taux_tunnel \leftarrow normaliser(exp(impact_grille_i + impact_grille_j));
- 7 carte_taux_tunnel \leftarrow fonction_normale_alatoire(carte_taux_tunnel, moyenne = déplacement_moyen, σ = écart-type);
- 8 Initialisation du signal modifié;
- 9 signal_modifié \leftarrow matrice_de_zeros(taille = taille(signal));
- 10 Déplacement des pixels;
- 11 **pour** $i \leftarrow 1$ à grandeur(signal, i) **faire**
- 12 **pour** $j \leftarrow 1$ à grandeur(signal, j) **faire**
- 13 $d \leftarrow$ arrondir(carte_taux_tunnel(i, j));
- 14 **si** signal = 1 & $j + d >$ grandeur(signal, j) & $d \geq 0$ **alors**
- 15 signal_modifié($i, j + d$) \leftarrow signal(i, j);
- 16 **fin**
- 17 **fin**
- 18 **fin**

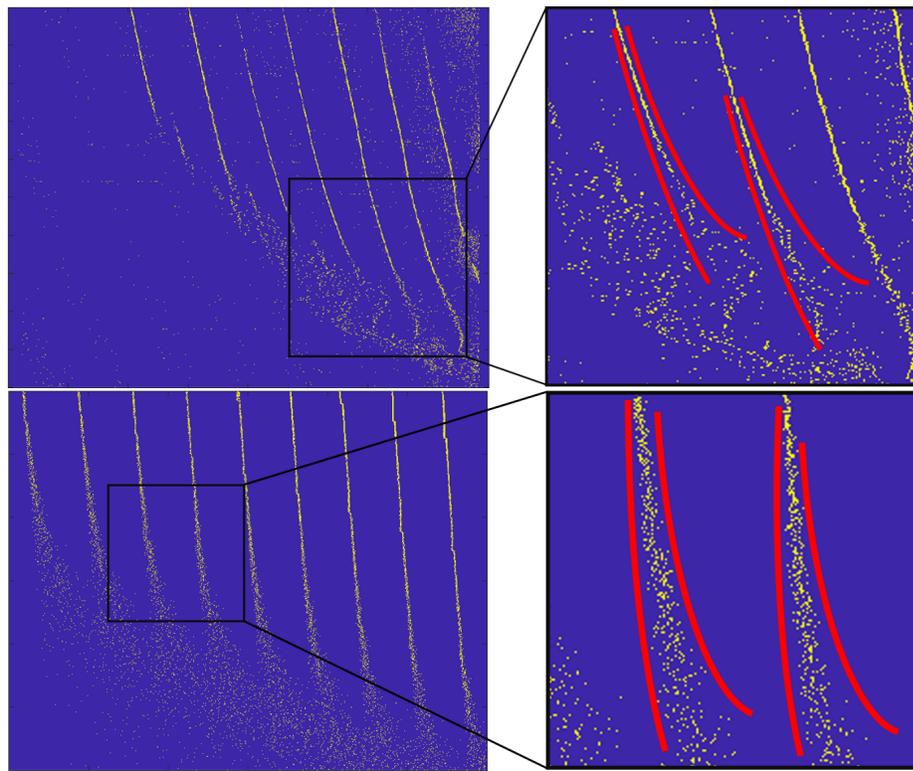
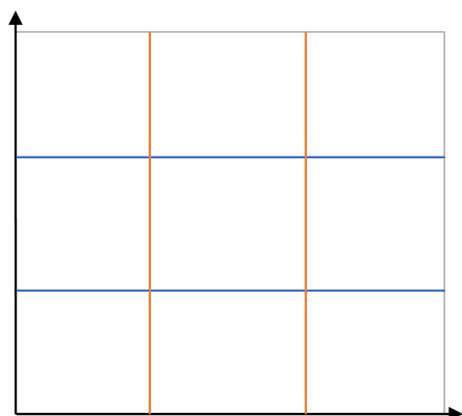


FIGURE 4.9 Exemple d'un diagramme de stabilité expérimental (haut) où la mesure était trop rapide, et d'une simulation (bas) modifiée par l'algorithme 11. Les lignes rouges délimitent l'impact de la mesure trop rapide.

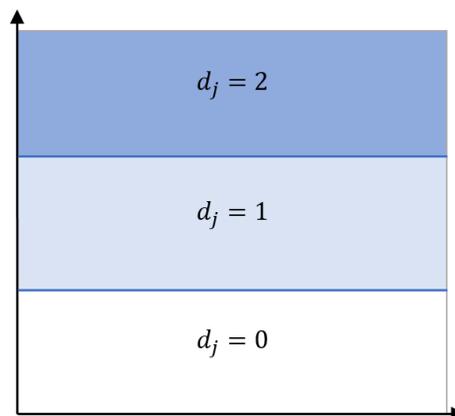
4.4 Effet de couplage dans un diagramme à double boîtes quantiques

De façon similaire à un défaut qui se couple à une boîte quantique due à sa proximité, deux boîtes quantiques peuvent être couplées ensemble et interagir de la même façon. Encore une fois, les deux effets du défaut couplé à la boîte quantique peuvent s'observer, soit la brisure de la ligne et l'anticroisement, et ce pour chaque croisement de chaque transition présente sur le diagramme. Un organigramme représentant les étapes suivies par l'algorithme est présenté à la figure 4.10.

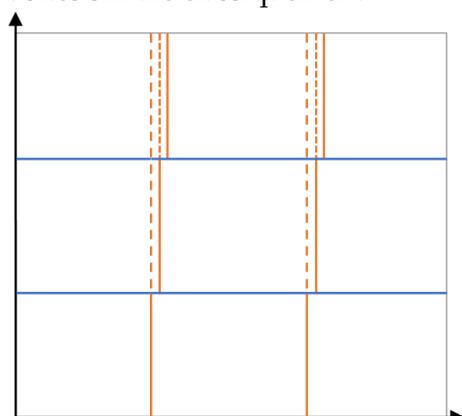
En plus de ces artefacts s'ajoute une ligne de signal reliant les sommets de deux hyperboles opposées. Pour créer cette dernière conséquence, il suffit d'élargir les lignes de transition jusqu'à ce qu'il y ait contact, puis de revenir à la largeur initiale en gardant les nouveaux contours créés. Cette suite d'étapes créera naturellement le lien entre les hyperboles, comme montré à la figure 4.11.



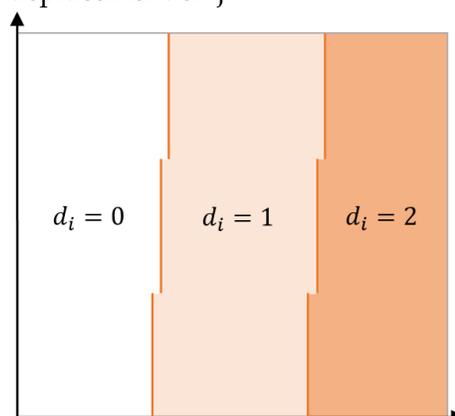
(a) Exemple : diagramme à double boîtes simulé classiquement.



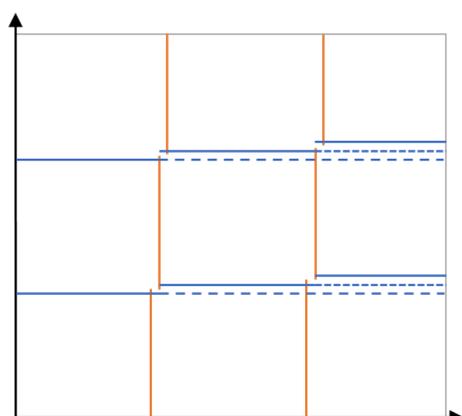
(b) Calcul des zones devant subir un déplacement en j .



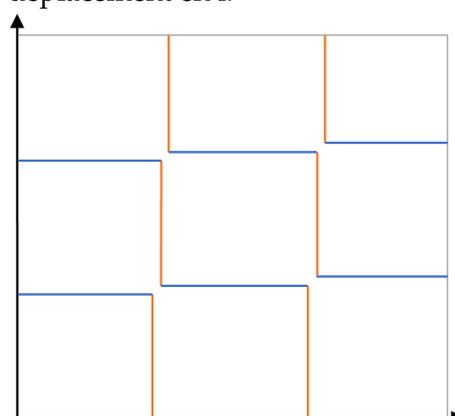
(c) Déplacement des pixels en j .



(d) Calcul des zones devant subir un déplacement en i .



(e) Déplacement des pixels en i .



(f) Résultat de la suite d'opérations et d'un lissage des lignes de transition.

FIGURE 4.10 Organigramme de l'algorithme ajoutant les effets de double boîte quantique au diagramme de stabilité.

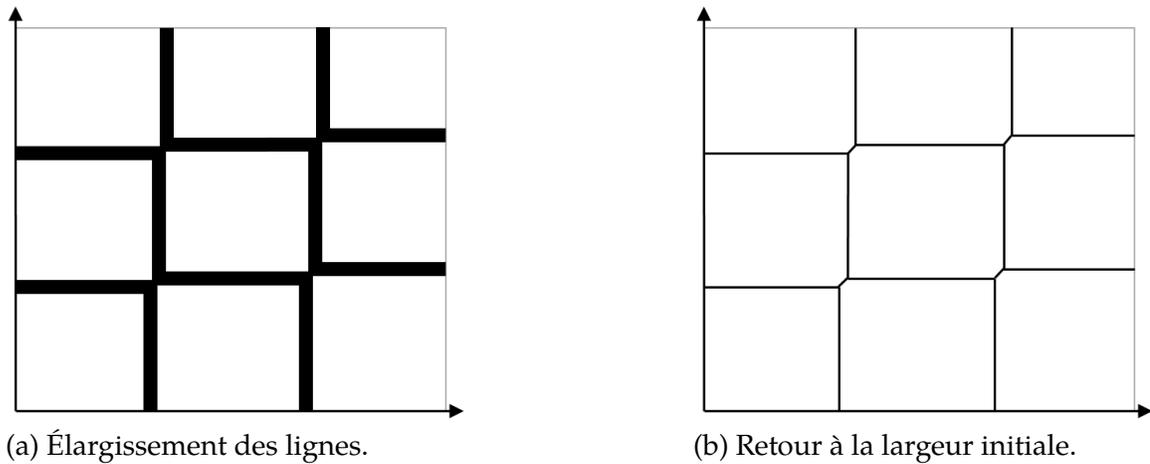


FIGURE 4.11 Étapes permettant l'ajout du signal aux points triples d'un diagramme de stabilité à double boîte quantique [2].

4.5 Effet du pré-traitement des diagrammes de stabilité expérimentaux

Comme mentionné au chapitre 1, les pixels étant identifiés comme une transition sur les diagrammes de stabilité expérimentaux sont ceux pour lesquels un retour de phase a pu être calculé sur le signal oscillant du TME. Cependant, pour pouvoir identifier ce retour de phase, il a fallu pouvoir mesurer une différence dans le signal, et donc en faire la dérivée. Un problème peut survenir sur le dos et le creux du signal oscillant, où la dérivée est presque nulle, rendant le transistor très peu sensible en ces points. Pour empêcher de confondre le bruit avec les modifications de l'occupation électronique, l'algorithme identifie automatiquement, à partir d'une certaine valeur, tous les pixels des creux et des dos comme négatifs. Ce choix permet d'éliminer un maximum de pixels faux-positifs, à défaut de quelques faux-négatifs. Sur le diagramme de stabilité, ce traitement se traduit par un effacement périodique des lignes de transition, comme illustré à la figure 4.12.

Pour reproduire cet effet, il faudra d'abord créer des oscillations qui dicteront les régions à effacer sur le diagramme de stabilité (en trouvant les creux et les dos). Un signal sinusoïdal est donc créé, représentant le signal du transistor mono-électronique qui pourrait être mesuré. Ce signal est ensuite modifié par une fonction de distorsion [37] [38], décrite à l'équation 4.2, qui permet de courber le signal, tel que pourrait l'engendrer des effets non linéaires d'accumulation de charges sous les électrodes (exemple expérimental à la figure 4.13, coin supérieur droit).

$$signal(i, j) = [a \cdot signal(i), b \cdot signal(j) - c \cdot \cos(signal(j))^d] \quad (4.2)$$

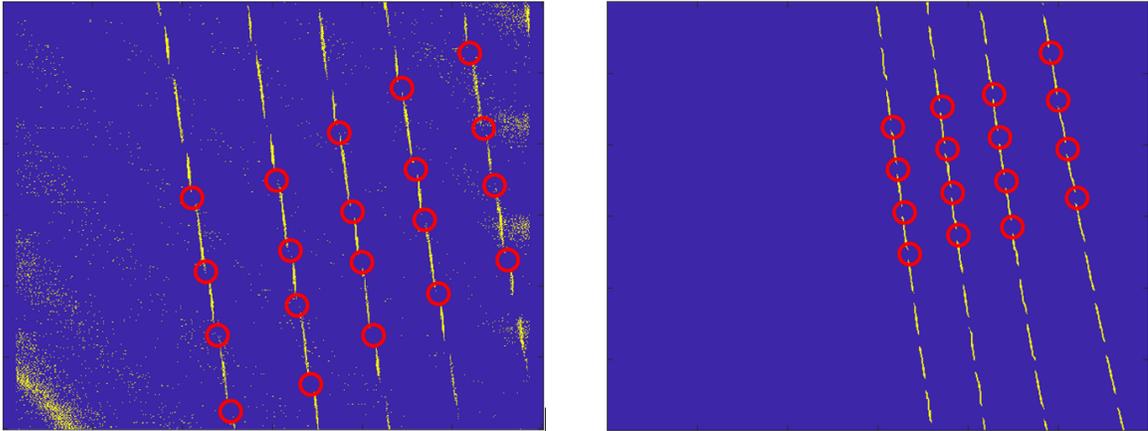


FIGURE 4.12 Exemple d'effacements périodiques de certaines parties des lignes de transitions sur un diagramme expérimental (gauche) et sur une simulation (droite), identifiées par des cercles rouges.

où *signal* est le signal sinusoïdal créé pour représenter une mesure de transistor, *i* et *j* sont respectivement les positions verticales et horizontales d'un pixel sur un diagramme, et *a*, *b*, *c*, *d* sont les paramètres de la distorsion.

Des exemples de signaux de transistors sont illustrés à la figure 4.13. Une fois cette carte créée, il suffit de la superposer à un diagramme de stabilité simulé, de choisir une valeur seuil sur le signal généré, et d'effacer les lignes de transition qui sont au-dessus de cette valeur choisie. Un exemple de l'application de cette transformation sur un diagramme simulé est montré à la figure 4.12.

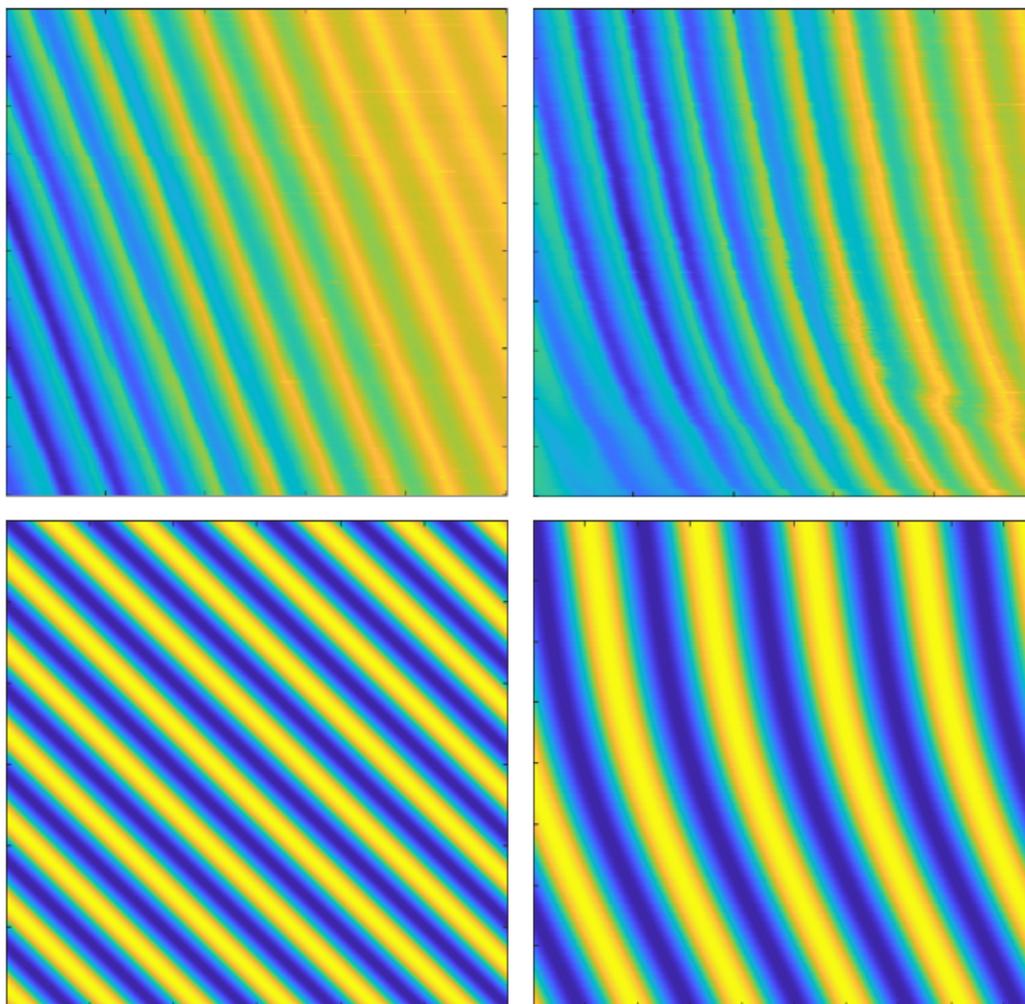


FIGURE 4.13 Exemples de signaux de transistors mono-électroniques mesurés (haut) et simulés (bas).

4.6 Ajout de pixels faux-positifs

Suite à ces différentes transformations, les diagrammes de stabilité générés sont considérés comme pouvant contenir tous les aspects physiques qu'un diagramme de stabilité expérimental peut contenir. Il manque toutefois un dernier ingrédient avant de pouvoir considérer les diagrammes générés comme représentation fidèle de la réalité. En effet, tous les diagrammes générés jusqu'ici sont uniquement composés de pixels vrais-positifs, c'est-à-dire qu'aucun pixel parasite n'est présent (les transformations appliquées sont comptées comme provenant d'un signal réel). Les prochaines sous-sections montrent deux techniques permettant de procéder à cet ajout.

4.6.1 Par distribution uniforme de bruit

La première technique ne consiste qu'en un ajout de pixels aléatoires et uniformes sur le diagramme de stabilité généré. Le but est simple : permettre à l'algorithme de reconnaître que certains pixels peuvent être mal identifiés, et ainsi être capable de cibler ceux provenant réellement d'une transition électronique.

Pour ce faire, une stratégie simple est utilisée :

1. Décider d'une valeur seuil qui déterminera le pourcentage de pixels à ajouter.
2. Tirer, au hasard, un nombre entre 0 et 1 pour chaque pixel du diagramme généré.
3. — Si le nombre est plus petit que le seuil choisi, le pixel sera changé en faux-positif ($0 \rightarrow 1$).
— Sinon, rien n'est fait.

Les diagrammes de stabilité ainsi générés comportent un certain nombre de pixels faux-positifs (exemples donnés à la figure 4.14 b et c), permettant idéalement à l'algorithme d'apprentissage de faire abstraction de certains pixels et d'améliorer la précision de son identification. L'avantage est donc de pouvoir obtenir de meilleures capacités d'apprentissage sans même avoir besoin d'informations sur les différents systèmes pouvant ajouter du bruit sur les données expérimentales. C'est d'ailleurs cette technique qui est utilisée dans le reste du projet, pour l'entraînement de l'algorithme d'apprentissage.

Si, toutefois, une banque de diagrammes de stabilité expérimentaux était disponible, il serait possible d'avoir recours à un algorithme d'apprentissage supplémentaire afin d'améliorer ou de créer une distribution de bruit plus représentative de ce qui est réellement observé.

4.6.2 Par apprentissage profond

La prochaine technique, bien que non utilisée pour l'entraînement de l'algorithme de segmentation, permettrait d'améliorer les performances d'identification de transitions électroniques en générant une distribution de bruit apprise à partir de données réelles. Le but étant toujours de générer des données qui ressemblent le plus possible à ce qui est mesuré en laboratoire, il en est de même pour l'ajout de pixels faux-positifs. Il y a cependant de nombreux éléments pouvant faire qu'un pixel est faussement identifié, allant de la fréquence et du moyennage utilisés pendant la mesure, aux instruments qui sont utilisés, en passant par les différents paramètres des algorithmes.

Pour résoudre ce problème, quoi de mieux qu'un algorithme statistique pour reproduire une distribution statistique. Il serait en effet possible d'utiliser un algorithme d'apprentissage pour apprendre la façon dont se dispersent les pixels faux-positifs dans un diagramme de stabilité. Pour ce faire, il suffit d'utiliser le même type d'algorithme que celui qui est utilisé dans le cadre de ce projet pour l'identification des transitions électroniques (décrit au chapitre 5), mais en lui assignant la mission inverse.

Initialement, le but était de créer des images connues grâce à un simulateur, d'entraîner un algorithme à « comprendre » ces images, puis à pouvoir identifier des données réelles grâce à cet entraînement. En inversant le sens de l'apprentissage, l'algorithme apprendrait à quoi ressemble un diagramme expérimental, puis tenterait de transformer la simulation dans ce style. En effet, en jouant avec les paramètres de l'algorithme, il est possible de jouer avec la taille des caractéristiques que celui-ci peut apprendre, et donc reproduire sur une image simulée. À ce moment, si les lignes de transitions simulées et modifiées par les techniques d'augmentation de données sont assez similaires à celles mesurées expérimentalement, la seule différence statistique entre les deux images est la distribution de bruit sur les données réelles. Entraîner l'algorithme à transformer une image simulée en image réelle ajouterait donc naturellement des pixels faux-positifs dans une distribution beaucoup plus près de la réalité.

L'avantage de cette technique est que la tâche d'entraînement peut se faire de façon non-supervisée (c'est-à-dire avec uniquement des images réelles, sans vérité terrain), grâce à des algorithmes comme *CycleGAN* [39] [40]. En effet, puisque les lignes de transitions générées sont considérées comme très près de la réalité, et qu'il ne suffit que d'ajouter quelques fausses données, le problème se résume à comprendre le comportement des pixels là où il n'y a pas de ligne. Ainsi, il devient possible d'apprendre la distribution de bruit directement à partir d'une image expérimentale, sans avoir besoin de comparaison avec quelconque autres identifications, souvent problématiques à se procurer.

Une fois entraîné, il est considéré que l'algorithme comprend la distribution de bruit d'un diagramme expérimental, et peut modifier une simulation selon cette distribution pour faire en sorte qu'elle ressemble davantage à la réalité. Pour ce faire, il y a deux façons d'y arriver, possédant chacune ses avantages.

1. Le diagramme à transformer ne contient que des lignes de transitions, et l'algorithme doit ajouter des pixels. Le problème à résoudre consiste donc à comprendre où se trouvent les pixels faux-positifs, et de les ajouter. L'avantage, dans ce cas, est qu'aucun paramètre initial n'est nécessaire.

2. Le diagramme à transformer contient déjà une distribution de bruit uniforme, où la densité est considérée comme la densité de bruit maximale en un point du diagramme expérimental (générée comme il est expliqué à la sous-section 4.6.1). La tâche de l'algorithme est donc d'enlever des pixels là où il y a normalement moins de bruit. L'avantage de cette seconde méthode est que la densité de bruit maximale est connue, rendant possible d'utiliser l'algorithme pour générer un plus grand ensemble statistique de données d'entraînement.

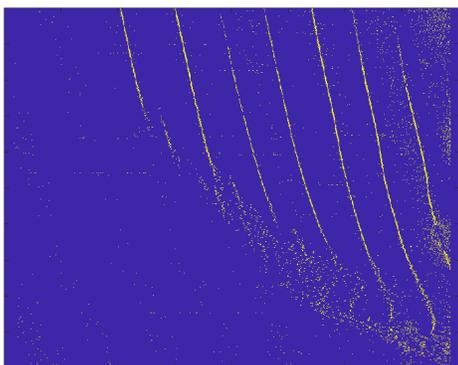
Le principe de fonctionnement étant bien différent, les algorithmes ne performant pas de la même façon pour les deux utilisations. En fait, aucun résultat concluant n'a pu être observé de la première façon. Une étude poussée de l'algorithme et des paramètres pourrait permettre de déterminer les causes d'erreur, ou même faire fonctionner la méthode, mais n'a pas été faite. La deuxième technique s'est cependant avérée plus concluante, et les résultats sont présentés à la figure 4.14.

Il est à noter que bien que la preuve de concept soit faite, l'utilisation de cette technique demeure peu fiable pour l'instant, étant difficile d'obtenir des résultats visuellement concluants. Il serait également utile de développer une façon d'évaluer si la distribution de bruit est bien reproduite sur la simulation. Toutefois, une fois testée et fonctionnelle, cette technique améliorerait sans doute les performances de l'identification du signal, et s'incorporerait bien dans un programme plus complet, en vue des perspectives décrites au chapitre 6.

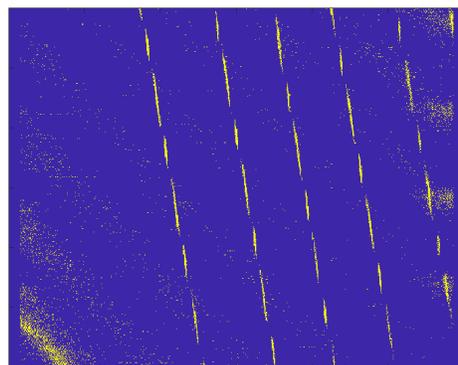
4.7 Combinaison d'artefacts et liste des paramètres

Les techniques de transformation présentées dans les sous-sections précédentes reproduisent aussi fidèlement que possible les artefacts qu'il est possible de retrouver sur un vrai diagramme de stabilité. Il est cependant évident que plus d'un artefact puissent être observés sur un diagramme. Il est donc primordial d'avoir une série de transformations indépendantes les unes des autres, afin de ne pas faussement introduire des caractères indésirables aux artefacts générés. Les transformations ont donc été implémentés de façon à pouvoir obtenir cette indépendance, à condition de les ajouter dans l'ordre suivant :

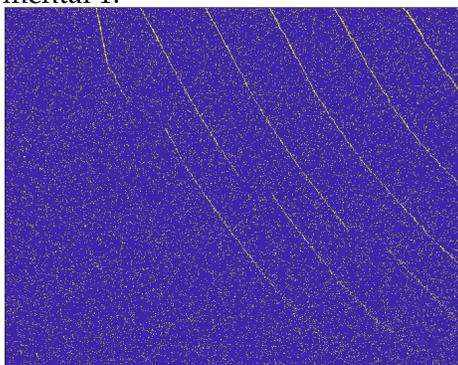
1. passage de l'occupation électronique à la mesure expérimentale ;
2. couplage dans un diagramme à double boîtes quantiques ;
3. défaut couplé au détecteur de charges et à la boîte quantique ;
4. défaut couplé au détecteur de charges, mais non à la boîte quantique ;



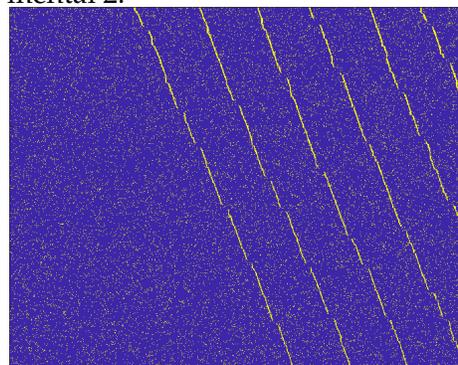
(a) Diagramme de stabilité expérimental 1.



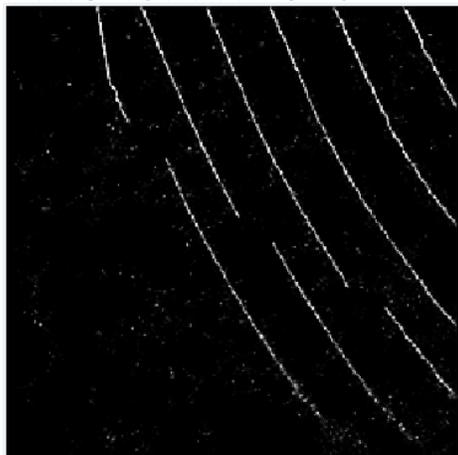
(b) Diagramme de stabilité expérimental 2.



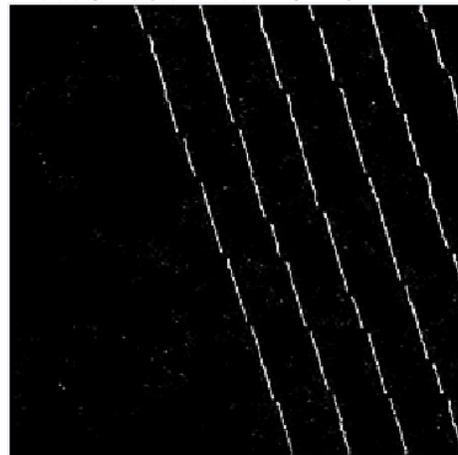
(c) Simulation 1 avec ajout d'une distribution de bruit uniforme.



(d) Simulation 2 avec ajout d'une distribution de bruit uniforme.



(e) Test de l'algorithme entraîné avec le diagramme de stabilité expérimental 1 sur la simulation 1.



(f) Test de l'algorithme entraîné avec le diagramme de stabilité expérimental 2 sur la simulation 2.

FIGURE 4.14 Exemple de diagrammes de stabilité utilisés pour l'entraînement et le test du *CycleGAN* pour l'ajout de bruit expérimental.

5. défaut couplé à la boîte quantique, mais non au détecteur de charges ;
6. transmission de la barrière tunnel ;
7. taux tunnel vs fréquence des mesures ;
8. effet du pré-traitement des diagrammes expérimentaux ;
9. ajout de pixels faux-positifs.

L'ordre énuméré ne découle pas tout à fait du hasard. En effet, il est important de suivre la causalité des évènements, c'est-à-dire qu'un défaut dans le dispositif s'est couplé à la boîte quantique avant qu'on la mesure, et la mesure s'est faite avant que le diagramme de stabilité ait été traité pour en extraire les lignes de transition. Ainsi, les éléments qui n'ont pas de lien de causalité entre eux n'ont techniquement pas d'ordre préférentiel, à l'exception de celui imposé par le défi de programmation (dans le cadre de ce mémoire, cet ordre-ci s'est montré indépendant et fonctionnel, mais aucun autre ordre n'a été testé).

Les étapes restantes, pour la génération d'une grande variété de diagrammes, sont de trouver les plages de validité des paramètres des différentes transformations, puis d'automatiser le processus de création d'une image. Il sera ainsi possible de générer un grand ensemble de données contenant suffisamment d'informations et d'artefacts pour entraîner un algorithme d'apprentissage à reconnaître de vraies données expérimentales.

Paramètre	Définition	Exemple de valeur(s)
<i>Transmission de la barrière tunnel</i>		
largeur_max	largeur maximale d'une transition	[2, 6] pixels
largeur_min	largeur minimale d'une transition	[1, 3] pixels
A	coefficient de couplage de l'abscisse	[1, 2]
B	coefficient de couplage de l'ordonnée	[1, 4]
n_moyennage	paramètre de lissage	[1, 15]
valeur_seuil	bruit interne des lignes	[0.95, 1]
<i>Taux tunnel vs fréquence des mesures</i>		
A	coefficient de couplage de l'abscisse	[4, 10]
B	coefficient de couplage de l'ordonnée	[1, 4]
prop_déplacement_moy	paramètre de déplacement des pixels	[0.01, 2]
prop_écart-type	bruit sur le déplacement	[0.01, 1]
<i>Effet du pré-traitement des diagrammes expérimentaux</i>		
A	coefficient de couplage de l'abscisse	[0.01, 0.05]
B	coefficient de couplage de l'ordonnée	[0.75, 2]
valeur_seuil	limite d'effacement de pixels	[0.85, 0.95]
a	coefficient de distorsion (éq. 4.2)	[-0.75, 0.85]
b	coefficient de distorsion (éq. 4.2)	[0.2, 0.7]
c	coefficient de distorsion (éq. 4.2)	[1, 2]
d	coefficient de distorsion (éq. 4.2)	[1, 5]
<i>Ajout de pixels faux-positifs</i>		
valeur_seuil	pourcentage pixels de bruits ajoutés	[0.5, 5] %

TABLE 4.1 Liste des distributions de paramètres d'entrée des différentes fonctions de traitement d'images utilisées pour la création de la base de données d'entraînement de l'algorithme de segmentation de diagrammes de stabilité.

Chapitre 5

Technique d'apprentissage profond de segmentation d'images pour l'identification de l'occupation électronique

« Une image vaut mille mots » ... à condition d'être capable d'en extraire l'information qui s'y retrouve. Cette tâche, généralement facile pour l'humain, peut s'avérer difficile pour un ordinateur. En effet, le simple fait de reconnaître une forme à travers un fond bruité requiert plusieurs étapes de traitements qui peuvent devenir très complexes avec l'augmentation de la taille de l'image, du nombre de couleurs et de formes possibles, et des nombreux autres paramètres augmentant le nombre de dimensions disponibles.

Une des techniques utilisées pour remédier à ce problème est la segmentation d'images, et une revue des différentes techniques peut être trouvée en [41]. Elle consiste à partitionner une image en différentes zones, permettant de faciliter l'extraction de l'information disponible. Une illustration simple de cette technique serait de transformer une image donnée en une peinture à numéro, où chaque couleur (numéro) possède une étiquette bien précise. Il s'agit donc d'avoir une transformation robuste qui permet, par exemple, d'identifier une forme recherchée dans un fond bruité, et de lui attribuer une étiquette particulière. Bien que cette technique soit robuste et efficace dans de nombreux domaines, le nombre de calculs exigé pour segmenter une image demeure relativement élevé, limitant donc son utilisation jusqu'à récemment.

En effet, avant les années 2000, un ordinateur n'était pas assez rapide pour entraîner un réseau de neurones performant dans un temps raisonnable [42] [43]. La segmentation d'images s'est donc longtemps limitée à des méthodes d'analyse basées sur les régions, les

contours, les groupements de pixels, ou sur des combinaisons de ces méthodes. Dans les années qui ont suivies, trois développements majeurs ont rendu possible l'utilisation des réseaux de neurones sur des problèmes de tous les jours [44]. Premièrement, l'arrivée de l'Internet et de l'infonuagique a permis le stockage et le transfert de grandes quantités de données. Deuxièmement, il a été possible d'enregistrer ces données localement grâce à l'augmentation de la capacité de mémoire. Et finalement, l'arrivée des processeurs graphiques a permis l'utilisation de ces données pour entraîner rapidement des réseaux neuronaux. Depuis, cette technique d'apprentissage a créé une réelle révolution dans le monde de la segmentation d'images, et gagne de nouvelles applications chaque jour.

5.1 La vision par ordinateur aujourd'hui

La vision par ordinateur, nom donné au domaine visant la compréhension d'images par un ordinateur, peut se diviser en quatre catégories [45] :

1. La classification d'images, qui permet d'attribuer une classe à une image. Elle est utilisée, par exemple, en foresterie pour la classification de feuilles d'arbre. Il s'agit, donc, étant donné une photo de feuille d'arbre, d'avoir un algorithme pouvant identifier à quelle essence d'arbre appartient la feuille. Elle peut également être utilisée pour déterminer si une feuille est malade ou non, et de la classer selon différentes maladies.
2. La détection d'objets, qui permet d'attribuer une classe à un groupe de pixels à l'intérieur d'une cadre. Ce type d'algorithme est utile, par exemple, pour la conduite automobile pilotée par ordinateur, où les voitures, personnes, animaux, et autres « classes d'objets » sont détectés et envoyés à l'ordinateur pour la prise de décision.
3. La segmentation sémantique (*semantic segmentation*), qui permet de différencier des objets de deux classes différentes dans une même image en attribuant une étiquette par pixel. Ce type de segmentation est très utilisé dans différents domaines de l'imagerie. Certaines entreprises s'en servent, par exemple, pour transformer des images satellites et carte étiquetée (carte routière, topographique, de développement urbain, etc) automatiquement. D'autres développements se font présentement en imagerie médicale, pour automatiser l'identification de cellules cancéreuses.
4. La segmentation de groupe (*instance segmentation*), qui permet de différencier des objets d'une même classe dans une image en attribuant une étiquette par pixel. Elle peut être utilisée, par exemple, en vidéo surveillance, pour identifier chaque personne différente se trouvant dans la scène filmée. Combinée à la segmentation

sémantique, elle tend également à remplacer la détection d'objets, ayant l'avantage d'une identification pixel par pixel, et non d'un cadre complet pouvant contenir des pixels de plusieurs classes.

Le choix de l'une ou l'autre de ces catégories, ou même une combinaison de celles-ci, peut dépendre de l'application choisie pour l'algorithme. Il s'agit donc d'opter pour une solution adaptée, en comprenant bien le problème ciblé et le fonctionnement général de ces techniques de vision par ordinateur.

5.2 Principe de fonctionnement des algorithmes de segmentation

Les algorithmes de segmentation étant plus généraux et polyvalents que de la classification d'images ou la détection d'objets, ces deux dernières méthodes seront mises de côté. Qui plus est, comme il sera montré à la prochaine section, il y a peu de différences entre la segmentation sémantique et celle de groupes. Il n'y aura donc pas de distinction algorithmique à faire entre ces deux méthodes, à l'exception de la différence entre les données à utiliser pour l'entraînement des algorithmes.

5.2.1 Données d'entraînement

Tout algorithme d'apprentissage a besoin de données d'entraînement à partir desquelles il pourra apprendre à réaliser la tâche souhaitée. C'est à ce moment que le choix de segmentation est fait. Si une segmentation sémantique est voulue, par exemple la transformation d'une image satellite en carte routière, une base de données d'images satellites et de cartes routières correspondantes doit être créée. De la même façon pour la segmentation de groupes, l'entraînement d'un algorithme pour l'identification de personnes sur vidéos de surveillance nécessite de créer une base de données avec des images réelles, ainsi que l'identification de chaque personne apparaissant sur ces images.

Puisque les réseaux de neurones sont récents et très complexes, et que leur fonctionnement est encore inconnu, il n'y a toujours pas de « règle d'or » pour la création de ces bases de données. Il s'agit d'y aller plutôt par essais et erreurs, et de trouver des solutions adaptées et personnalisées au problème à résoudre. Il y a cependant deux idées importantes à garder en tête lors de la création de la base de données :

1. Plus les images ont de dimensions (grande taille, plus de canaux de couleurs, etc), plus la base de données devra être grande.

2. Plus la capacité de l'algorithme (nombre de neurones, de couches de neurones, etc) est grande, plus la base de données devra être grande.

Il est ainsi possible de jouer avec le nombre de dimensions des données d'entraînements et la capacité de l'algorithme pour l'optimisation de ce dernier.

5.2.2 Algorithme et entraînement

Le type d'algorithme utilisé, et décrit ci-dessous, se nomme *GAN* («*generative adversarial network*») [39] [40] [46]. Ces algorithmes se distinguent par un fonctionnement très particulier, où non un, mais deux réseaux de neurones sont utilisés, et entrent en compétition. La base de données créée est d'abord définie, en nommant « images réelles » les images observées par l'instrument de mesure (par exemple l'image satellite), et « vérités terrains » les images étiquetées (par exemple la carte routière). Il y a donc un premier réseau, nommé *générateur*, qui tentera, à partir d'une image réelle, d'apprendre à générer la vérité terrain correspondante. Un deuxième réseau, nommé *discriminant*, tentera de faire la différence entre la vérité terrain générée par le générateur et celle de la base de données d'entraînement.

Évidemment, lors des premières itérations de l'algorithme, le générateur n'a pas appris à faire la transformation de l'image réelle vers la vérité terrain. Le discriminant peut donc facilement faire la différence, et distinguent la vérité terrain de l'image générée avec une erreur d'environ 0 %. Au fur et à mesure que les itérations avancent, le générateur apprend à bien générer la vérité terrain. Le discriminant voit donc son erreur augmentée, n'étant plus capable de différencier à coup sûr la vérité terrain générée de la vraie, provenant de la base de données d'entraînement. L'algorithme poursuit son apprentissage jusqu'à ce que l'erreur du discriminant converge à 50 %. À ce moment, le générateur est considéré comme pouvant générer des vérités terrain tellement près de la réalité qu'il peut être utilisé pour créer lui-même de nouvelles vérités terrains sur des données qu'il n'aurait même jamais vues pendant l'entraînement.

5.3 Application sur les diagrammes des stabilité

De la même façon qu'une image satellite est transformée en carte étiquetée pour en faciliter l'interprétation, le but de l'algorithme à développer ici est de permettre à un ordinateur d'identifier et de différencier les lignes de transitions sur un diagramme de stabilité. Ces tâches consisteront donc à enlever les pixels positifs bruyants, et à étiqueter les autres de façon à ce qu'il soit possible d'extraire la position de chacune des différentes lignes. Pour ce faire, il faudra d'abord trouver une façon de créer la vérité terrain d'un diagramme de stabilité, puis de générer une base de données suffisamment grande pour l'entraînement de l'algorithme d'apprentissage. Il sera ensuite possible de le tester sur des données mesurées en laboratoire et d'en valider l'identification.

5.3.1 Génération de la vérité terrain

Bien qu'il soit en général difficile de se procurer de bonnes données étiquetées, l'utilisation d'un simulateur et de techniques de traitement d'images en facilite grandement la création. En effet, pour ce qui est de la deuxième tâche d'identification et de différenciation des lignes de transition, il suffit de partir de l'occupation électronique calculée suite à la simulation, et d'avoir un programme exécutant les étapes suivantes :

1. Créer le signal de détection de charges grâce à l'algorithme 5, et le transformer en valeurs booléennes ($0 \rightarrow \text{faux}$, $1 \rightarrow \text{vrai}$).
2. À partir de la matrice de booléens créée, si l'élément de matrice du pixel de l'occupation électronique correspondant est :
 - *faux*, le pixel de vérité terrain vaudra 0;
 - *vrai*, le pixel de vérité terrain sera égal à la moyenne des huit pixels autour de celui-ci, arrondie à l'entier le plus haut.
3. La vérité terrain est finalement normalisée pour s'assurer que la première ligne observée soit égale à 1, et ainsi de suite.

À partir de ce moment, la dérivée d'une occupation électronique et sa vérité terrain correspondante font la paire et suivront les mêmes étapes de transformation d'images. En effet, comme présenté au chapitre 4, les images passeront par une suite de techniques d'augmentation de données pour améliorer leur réalisme. Heureusement, ces dernières ne font que déplacer les pixels déjà en place, ou se servir de ceux-ci pour en ajouter de nouveaux. Il est donc possible de modifier le diagramme de stabilité et sa vérité terrain en

même temps, et de garder la validité de l’identification des lignes de transition malgré leurs modifications et l’ajout d’artefacts.

Finalement, pour ce qui est de la tâche d’enlever les pixels bruyants, il suffit de modifier uniquement le diagramme de stabilité, et non sa vérité terrain, lors des étapes d’ajout de pixels de bruit. La paire d’images ainsi générée peut donc être utilisée à titre d’entraînement pour l’algorithme, lui donnant l’information dont il a besoin pour réaliser les tâches d’élimination des pixels bruyants et d’identification des lignes de transition.

5.3.2 Création de la base de données d’entraînement

Encore une fois, l’utilisation d’un simulateur et de techniques de traitement d’images facilitent grandement cette tâche. Pour ce qui est du nombre de données d’entraînement, ou de la variété des défauts à ajouter aux diagrammes de stabilité, il ne s’agit ici pas d’une science exacte, ou du moins pas pour l’instant. En effet, les réseaux de neurones et leur fonctionnement sont des domaines récents et encore à l’étude. Ainsi, il est difficile de connaître leur comportement ou leur taux de succès autrement qu’en testant d’abord puis en validant. C’est pourquoi les caractéristiques de la base de données générées, comme le nombre de diagrammes, de défauts ou encore de balayage en potentiel, relèvent ici plus de l’essai-erreur que de la science. Un total de 600 diagrammes de stabilité et vérités terrains ont donc été générés selon les critères suivants (exemples à la figure 5.1) :

- 3 dispositions de grilles différentes ont été choisies pour créer 3 dispositifs, avec toutefois les mêmes paramètres physiques ;
- 4 balayages en potentiel par dispositif, créant 12 occupations électroniques différentes ;
- 5 combinaisons d’artefacts aléatoires par balayage, créant 60 diagrammes de stabilité différents ;
- 10 plus petites parcelles ont été choisies par diagramme, à des positions aléatoires, créant 600 diagrammes de stabilité.

Afin de réaliser chacune des étapes décrites plus haut, les paramètres des tableaux 3.1, 3.2 et 4.1 ont été utilisés (validés par inspection visuelle, selon les critères décrits aux sections correspondantes). Pour ce faire, un programme ayant en mémoire chacun de ses paramètres répétait chaque étape en lançant au hasard les paramètres désirés variables, et en les envoyant au simulateur ou aux fonctions de traitement d’images. Il s’agit donc de choisir à l’avance les plages de validité de chacun de ses paramètres, et d’exécuter un algorithme qui peut générer les diagrammes de stabilité selon les étapes voulues.

Une fois cette banque créée, les images peuvent être utilisées pour entraîner l’algorithme,

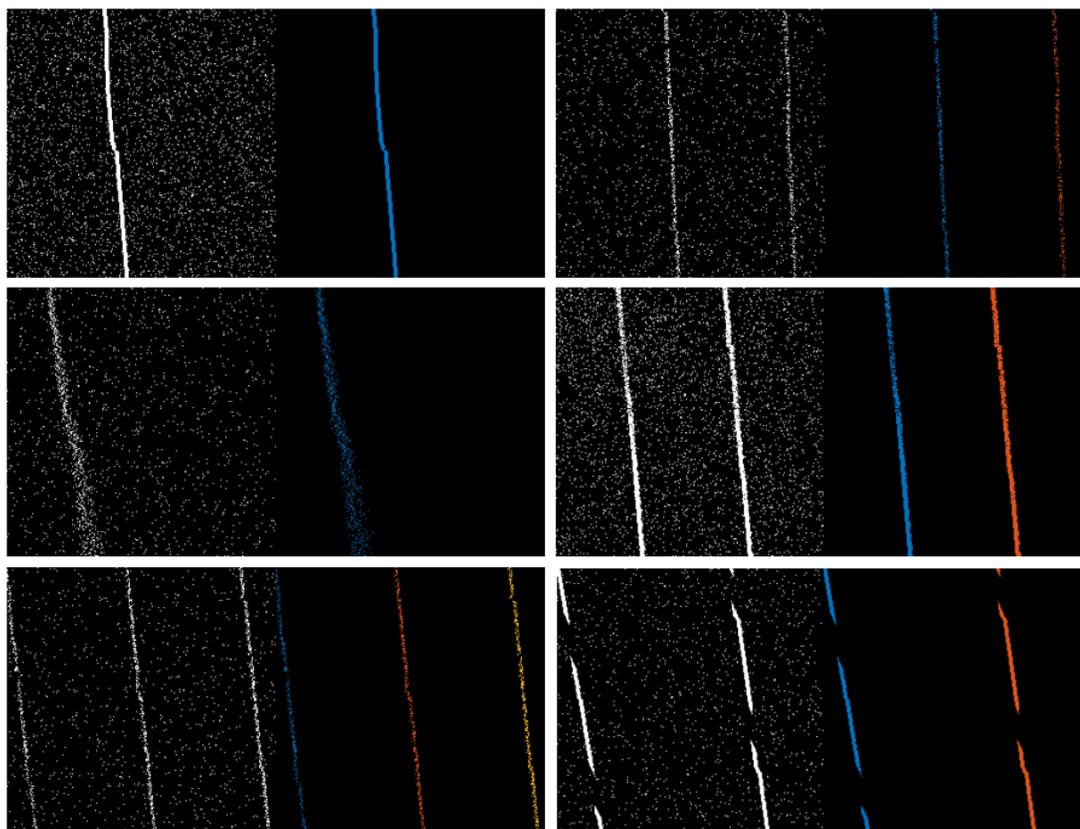


FIGURE 5.1 Exemples de diagrammes de stabilité simulés (à gauche pour chaque image) et leurs vérités terrains (à droite) générés pour l'entraînement de l'algorithme.

qui pourra ensuite être testé sur de vrais diagrammes expérimentaux pour en tester la performance. Dans le cas de cette étude, seule une banque de 600 diagrammes a été générée, et directement testée.

Il est à noter que le nombre d'images utilisées ici est relativement faible comparé à ce qui se fait, et général, avec ce type d'algorithme (pouvant atteindre des millions d'images). Dans le futur, un plus grand nombre de diagrammes de stabilité pourrait être utilisé pour l'entraînement, permettant possiblement d'augmenter les performances et la généralité de l'identification de l'algorithme. Cependant, n'étant qu'à l'étape de preuve de concept, il serait trop tôt pour se lancer dans des études de performances et de taille de base de données. En effet, il pourrait être utile de connaître l'ordinateur utilisé en laboratoire pour l'entraînement de l'algorithme, la mémoire disponible, les performances de réussites visées, le temps de calcul souhaité (fait en temps réel ou calculé à l'avance?), et autres caractéristiques importantes du programmes souhaités avant d'en former la base de données d'entraînement. Ainsi, pour la preuve de concept, un nombre relativement faible de diagrammes est utilisé,

permettant de manipuler et de valider rapidement chacune des étapes, tout en étant suffisant pour l'apprentissage et les tests à effectuer.

5.3.3 Paramètres de tests et résultats

Afin de tester l'algorithme, un certain nombre de diagrammes de stabilité acquis il y a quelques années ont été utilisés. Étant pris sur des dispositifs et balayages en potentiel différents, il sera ainsi possible de montrer la généralité de l'algorithme. Il est cependant à noter que les finalités du projet s'arrêtent à la preuve de concept du fonctionnement de la méthode. Aucun résultat quantitatif sur la performance de l'identification n'a donc été recueilli. Il s'agit ici simplement de démontrer qu'il est possible d'identifier avec précision les différentes lignes de transition sur un diagramme de stabilité expérimental, et ce, même si elles sont affectées par différents artefacts réels.

En ce qui concerne l'ordinateur utilisé, les calculs ont été lancés sur l'ordinateur *BELUGA* (Calcul Canada), avec 1 processeur graphique, 6 processeurs centraux, ainsi que 32Go de mémoire par noeud. Le temps nécessaire à l'entraînement et aux tests était d'environ 2 minutes. Pour ce qui est de l'algorithme *pix2pix* [39] [40], les paramètres par défauts ont été utilisés, ainsi que les entrées suivantes :

- *input_nc* et *output_nc* (nombre canal de couleur des données d'entrée/sortie = 3 ;
- *load_size* et *crop_size* (taille NxN des images chargées/à traiter) = 256

Les résultats, dont certains exemples concluants sont présentés à la figure 5.2, montrent qu'il est possible d'avoir une identification pixel à pixel d'un diagramme de stabilité expérimental faite grâce à un algorithme d'apprentissage entraîné uniquement à partir de données simulées. En effet, comme il est possible de le juger visuellement, les lignes entières ont pu être identifiées de façon robuste, malgré la courbure, les effacements périodiques, le bruit télégraphique, et autres artefacts présents.

En ce qui a trait au taux de succès, seule une brève évaluation a été faite. Il est d'abord utile de rappeler les deux buts principaux de l'algorithme, soit de retirer les pixels bruyants du diagramme de stabilité, et d'en distinguer les lignes de transition. L'évaluation de ceux-ci se fait par inspection visuelle, sur un ensemble de test de 50 diagrammes de stabilité expérimentaux. Pour ce qui est du premier, un taux de réussite d'environ 90% a pu être observé, alors qu'un taux d'environ 50% a été observé pour la coloration des lignes de transition. Bien que faible, ce dernier taux n'est pas alarmant. En effet, il est important de rappeler qu'un réseau de neurones débute son apprentissage en ne pouvant identifier aucun diagramme de stabilité. Si l'apprentissage n'avait pas fonctionné, une image bruitée

aurait été retournée, n'ayant aucune signification. Le simple fait de retrouver des résultats concluants est donc la démonstration d'un apprentissage réussi. Dans le futur, il serait toutefois important de maximiser ces taux de réussite, comme décrit dans le prochain chapitre sur les perspectives de la méthode.

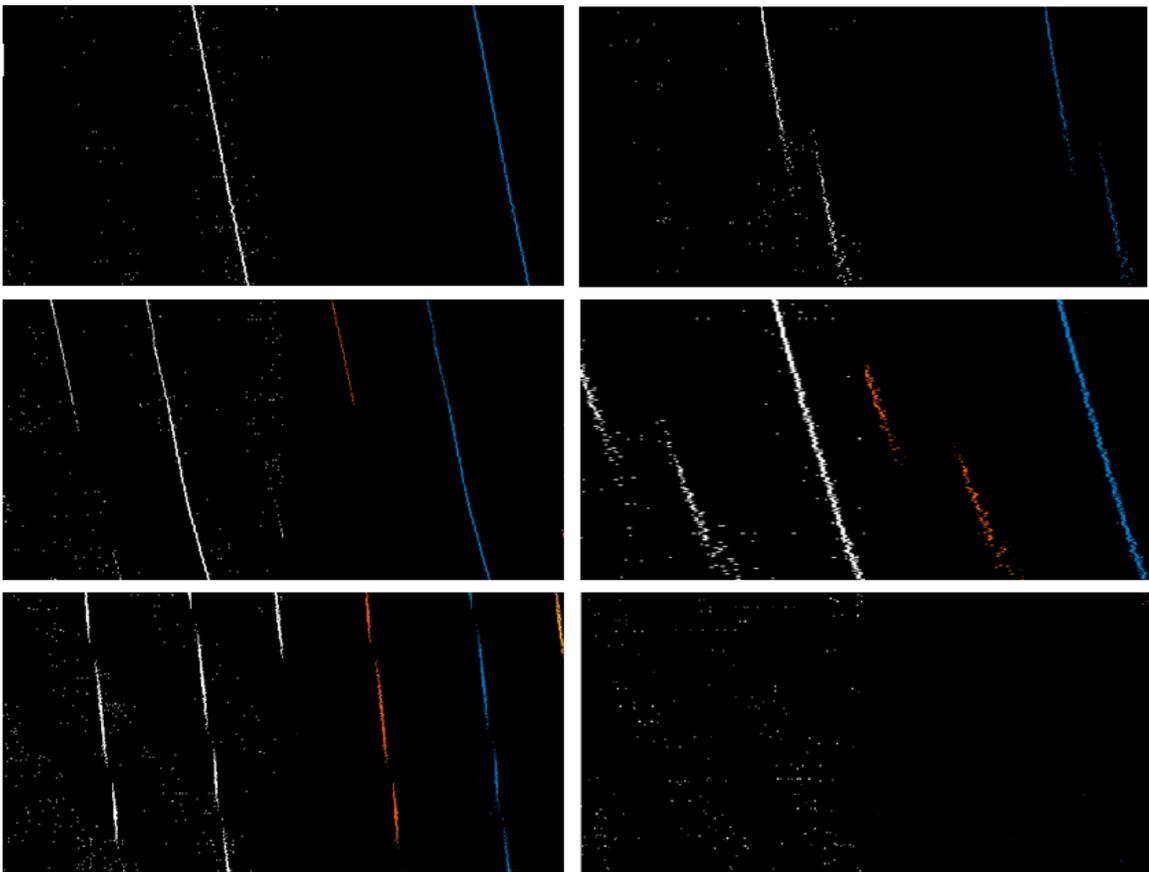


FIGURE 5.2 Exemples de diagrammes de stabilité expérimentaux (à gauche pour chaque image) et leurs identifications faites par l'algorithme de segmentation (à droite). À noter que le diagramme du coin inférieur droit est un exemple de diagramme expérimental sans ligne de transition, montrant l'identification où aucune ligne de transition n'est coloriée, mais où tous les pixels de bruit ont été retirés, laissant l'image vide.

Chapitre 6

Perspectives

Les résultats précédents montrent qu'il est possible, à partir d'un modèle physique simple et de techniques d'augmentation de données, d'entraîner un algorithme de segmentation permettant l'identification des transitions électroniques sur un diagramme de stabilité expérimental. En effet, comme mentionné au chapitre 5, l'entraînement de l'algorithme d'identification de signal n'a nécessité aucune donnée expérimentale. Ceci montre que, pour pouvoir identifier les lignes de transition d'une boîte quantique sur un diagramme de stabilité mesuré par un TME, il suffit d'une légère connaissance du système qui sera utilisé (dispositif, montage et pré-traitement expérimental), ainsi que de quelques fonctions de traitement d'images. En d'autres mots, aussi compliqué que coincer un seul électron dans un boîte quantique à l'intérieur d'un dispositif cryogénisé puisse être, s'il est possible de bien dessiner une ligne, il est envisageable de le faire.

Ces avancées, bien que prometteuses pour l'automatisation complète et robuste de l'initialisation des boîtes quantiques dans le régime à un électron, nécessiteraient maintenant une large phase de tests en ce qui concerne l'algorithme d'apprentissage. Les sections suivantes présenteront donc, en premier lieu, l'idée générale concernant les tests qu'il faudrait faire pour optimiser la segmentation des lignes de transitions. Par la suite, deux idées de continuité seront développées, montrant la grande généralité et les nombreux avantages qu'apportent l'utilisation d'un algorithme d'apprentissage profond entraîné à partir de simulations.

6.1 Optimisation de l'algorithme

L'algorithme de segmentation et les résultats présentés démontrent que la technique d'identification de signal peut fonctionner. Il est cependant important de ne pas tirer davantage de conclusions sur les performances de l'algorithme. En effet, de nombreux autres algorithmes de segmentation existent, ou sont actuellement en développement, et permettraient eux aussi l'identification des lignes de transitions. Ils ont, évidemment, tous leurs avantages et inconvénients, et leurs listes de paramètres bien à eux. Une étude exhaustive des algorithmes et de leurs paramètres permettrait de déterminer l'algorithme le plus performant, et de fournir des résultats quantitatifs sur ses performances.

Une fois cette tâche réalisée, il sera possible de déterminer l'impact des différents choix faits lors de l'élaboration de la base de données d'entraînement, et de les optimiser à leur tour. Parmi ces choix se trouvent le nombre de diagrammes de stabilité à fournir en entraînement, le nombre de dispositifs simulés permettant la génération de ceux-ci, le type de bruits (faux-positifs) ajoutés lors de l'augmentation de données, et toute autre caractéristique ou paramètre jouant un rôle dans l'apprentissage de l'algorithme.

Ces étapes d'optimisation de l'algorithme et de sa base de données d'entraînement transformeraient la segmentation de diagrammes de stabilité en réel outil expérimental, testé et optimisé quantitativement. La suite du chapitre explique comment il serait possible d'étendre l'utilisation de l'algorithme de segmentation.

6.2 Multidimensionnalité de l'algorithme

Comme démontré dans la section sur la création de la base de données d'entraînement, seuls des diagrammes de stabilité de dispositifs à une boîte quantique, créés par le balayage en potentiel de deux grilles seulement, ont été générés ou identifiés. Ayant en main un algorithme n'étant pas limité en dimensions, il serait possible d'étendre l'utilisation de cette technique, soit pour détecter un signal provenant de plusieurs boîtes quantiques, ou encore pour des diagrammes de stabilité à plus de deux dimensions.

6.2.1 Expansion à plusieurs boîtes quantiques

Étant un algorithme de segmentation basé sur la comparaison entre une image réelle et son identification voulue, et pouvant générer tout type de diagramme de stabilité, rien ne semble empêcher l'identification de signaux provenant d'autant de boîtes quantiques que possibles. C'est effectivement la seule limite en ce qui a trait à l'identification d'un diagramme expérimental : avoir la fonction de traitement d'images nécessaire pour créer l'identification voulue.

Il y a cependant une seconde limite, expérimentale cette fois, à cette détection de multiples boîtes. En effet, plus une boîte quantique est située loin du TME au moment de la mesure, plus le signal de ses transitions électroniques est faible, pouvant disparaître complètement si elle trop loin. Ceci se traduit, sur le diagramme de stabilité expérimental après traitement, par une différence en proportions de pixels vrais-positifs en fonction de la distance de la boîte quantique au TME.

Il semble donc, intuitivement, que ce problème est d'une plus grande complexité pour un algorithme de segmentation, pouvant créer des problèmes de performances. Différentes stratégies pourraient être utilisées pour y remédier, comme la modification du prétraitement expérimental de Maxime Lapointe-Major [26], en augmentant la dimension des données d'entraînement (trois valeurs de pixels possibles donnant un indice sur l'intensité du signal par exemple). En d'autres mots, ayant déjà un bon outil d'identification des lignes de transition, il serait possible d'adapter la mesure et la base de données d'entraînement afin de rendre possible et d'améliorer l'expansion de la segmentation à plusieurs boîtes quantiques.

6.2.2 Expansion à plusieurs grilles de contrôle

Une deuxième dimension qu'il serait possible d'augmenter serait le nombre de grilles de potentiel utilisées pour le balayage. En effet, les techniques utilisées jusqu'ici étaient limitées à une utilisation de diagrammes de stabilité en deux dimensions. Les réseaux de neurones n'ont cependant pas cette limitation, et la technique pour créer la base de données d'entraînement non plus. Il serait donc possible, encore une fois à condition d'avoir la technique d'augmentation de données nécessaire, d'apprendre à un algorithme à identifier un signal à plusieurs dimensions.

Une telle utilisation augmenterait énormément l'indépendance de l'algorithme, permettant une recherche de signal sur un espace de possibilités de potentiels beaucoup plus grand, maximisant du même coup la chance de trouver la dernière transition électronique. En effet,

si un dispositif à trois grilles est utilisé, et que des diagrammes à deux dimensions seulement sont mesurés, un minimum de trois diagrammes sont nécessaires pour mesurer chacun des couplages entre les grilles et la boîte quantique. Si des mesures en trois dimensions sont faites, il serait possible de faire le tout en une seule mesure, et de choisir le régime adéquat beaucoup plus rapidement.

6.3 Amélioration en temps réel de la performance de l'algorithme

Une des clés du bon fonctionnement d'un algorithme d'apprentissage se trouve dans ce qui est donné à ce dernier comme apprentissage. En effet, le nombre de données d'entraînement, leur variété et leur validité sont des exemples de caractéristiques très importantes à contrôler pour maximiser la performance de l'algorithme. Puisqu'il est facile de contrôler et de générer à volonté les données d'entraînement de l'algorithme de segmentation utilisé, il serait possible de les adapter au fur et à mesure que des informations sont obtenues sur le dispositif mesuré.

6.3.1 Modification de la base de données d'entraînement

La base de données générée comporte d'abord un ou plusieurs types de défauts ajoutés par augmentation de données, puis une certaine quantité de ces dits défauts. Il serait donc utile, si le dispositif mesuré est plus ou moins connu en début d'initialisation, d'avoir le plus large éventail de types et de nombre et de défauts, maximisant la probabilité d'identifier une transition électronique sur la mesure. Ensuite, si pendant l'initialisation du dispositif, il est remarqué qu'un type de défaut n'apparaît jamais, il serait possible d'entraîner de nouveau l'algorithme de segmentation sans ce dernier type, augmentant la performance d'identification pour le dispositif mesuré. Il serait, de la même façon, possible de modifier certains paramètres ou nombre de défauts, toujours dans l'objectif de faire ressembler le plus possible la base de données générée aux diagrammes de stabilité mesurés.

6.3.2 Ajout de données expérimentales à la base de données d'entraînement

Si l'idée est de faire ressembler le plus possible la base de données générée aux diagrammes de stabilité mesurés, pourquoi ne pas ajouter des diagrammes de stabilité expérimentaux directement dans la base de données d'entraînement ? C'est effectivement l'idée qui est proposée ici. Le problème réside dans la création de la vérité terrain du diagramme de stabilité expérimental. Pour le résoudre, il faudrait être capable de valider si l'identification faite par l'algorithme est concluante, et si celle-ci peut servir de donnée d'entraînement. Une fois cette validation faite, deux choix d'utilisation s'offrent :

1. Mettre à jour la base de données immédiatement, avec cette nouvelle donnée expérimentale, et entraîner l'algorithme à nouveau. Ainsi, chaque nouvelle mesure de signal augmenterait la chance de trouver la prochaine.
2. Utiliser les données d'une initialisation précédente pour créer une base de données d'entraînement propre à chaque dispositif (ou combinaison de grilles de potentiel), contenant des diagrammes expérimentaux identifier de ce même dispositif (ou combinaison de grilles). Ainsi, il serait possible d'entraîner un algorithme de segmentation de façon personnalisée, en fonction du dispositif et des grilles de potentiel utilisées pour faire la mesure.

Le but de cette approche est d'améliorer les performances de la reconnaissance. En effet, l'ajout de données expérimentales dans la base de données d'entraînement permettra sans doute à l'algorithme de faire une meilleure étude de ce qu'il aura prochainement à identifier. Cet ajout permettra donc, en premier lieu, de réussir un plus grand nombre de prédiction (nombre de vérités terrain prédites avec succès étant donné plusieurs diagrammes de stabilité expérimentaux). Ensuite, puisque l'algorithme est entraîné avec des données réelles, le nombre de pixels identifié avec succès sur un diagramme de stabilité a de fortes chances d'augmenter lui aussi, permettant de faire une meilleure analyse et prise de décision pour un algorithme ayant à décidé de prochaines plages de potentiels à explorer.

Conclusion

Ce travail consiste en la preuve de concept d'un algorithme d'apprentissage pouvant faire la segmentation, pixel à pixel, d'un diagramme de stabilité expérimental, entraîné uniquement à partir de données générées. Ces résultats sont prometteurs pour l'initialisation automatique dans un régime utile, montrant une prédiction précise et robuste aux bruit et artefacts expérimentaux des lignes de transition sur un diagramme de stabilité d'une boîte quantique unique.

Bien que nécessitant d'abord une optimisation et une qualification quantitative avant l'instauration en laboratoire, la méthode proposée permettrait l'échelonnabilité de la technologie en facilitant la généralisation à plusieurs boîtes quantiques. De plus, le fait d'avoir un algorithme de reconnaissance pixel à pixel permettrait d'augmenter le nombre de dimensions balayées pendant la mesure, et d'ainsi explorer un plus vaste espace de potentiel pour trouver le régime voulu.

L'utilisation du modèle de Thomas-Fermi pour l'implémentation du simulateur a permis de générer rapidement et facilement des diagrammes de stabilité physique. Afin d'ajouter les artefacts qui peuvent être observés sur un diagramme de stabilité expérimental, mais qui ne sont pas pris en compte par le modèle, des techniques de traitement d'images ont été utilisées. Celles-ci permettent l'augmentation de données nécessaire à l'entraînement d'un algorithme d'apprentissage.

Ce dernier permet de retirer les pixels bruyants d'un diagramme de stabilité expérimental, et d'en distinguer les lignes de transition électroniques, s'il y a lieu. Une simple recherche des pixels non-nuls permet l'identification d'une ligne indépendante, qui pourrait être envoyée à un algorithme de prise de décision, automatisant le processus d'initialisation.

Le développement de ces dernières techniques sont des approches nouvelles dans le domaine des boîtes quantiques sur dispositif semi-conducteur, permettant de faire une identification automatique et intelligente de son occupation électronique. De plus, elles

consistent en de nouvelles applications des réseaux de neurones et en une façon originale de création de base de données d'entraînement, augmentant le champ d'expertise du domaine de l'intelligence artificielle. La continuité de ce projet, décrite au chapitre sur les perspectives, se voudrait le développement d'un module d'identification robuste, ayant la capacité d'améliorer lui-même ses performances, et permettant l'augmentation de la dimensionnalité du problème.

Pour conclure, un algorithme d'apprentissage a su identifier le signal provenant de la modification de l'occupation d'une boîte quantique, étant entraîné uniquement à partir de données créées et dessinées de toutes pièces. Cette réalité amène une nouvelle problématique en ce qui concerne la résolution d'un problème physique : devons-nous réellement apposer une équation, à priori approximative car aucune équation universelle n'existe encore, sur chaque élément d'un système physique pour le comprendre ? Ou pouvons-nous demander à un algorithme d'apprentissage, possédant une complexité algorithmique et dimensionnelle si grande qu'il est impossible de se l'imaginer, de faire cette compréhension pour nous grâce à un apprentissage stratégique ? Il est à noter que cette réalité interfère déjà, depuis quelques années, en physique. En effet, dans le domaine de la supraconductivité, des algorithmes d'apprentissage peuvent prédire des alliages favorisant certaines caractéristiques [47] [48] [49] [50], permettant à des expérimentateurs de se concentrer sur la fabrication des matériaux.

Bibliographie

- [1] Lapointe-Major, M. Automatisation de l'initialisation d'une boîte quantique au régime à un électron. *mémoire de maîtrise* (2018).
- [2] Hanson, R., Kouwenhoven, L. P., Petta, J. R., Tarucha, S. & Vandersypen, L. M. K. Spins in few-electron quantum dots. *Reviews of Modern Physics* **79**, 1217–1265 (2007). URL <http://dx.doi.org/10.1103/RevModPhys.79.1217>.
- [3] Schaller, R. R. Moore's law : past, present and future. In *IEEE Spectrum*, vol. 34, 52–59 (June 1997).
- [4] Kumar, S. Fundamental limits to moore's law (nov. 2015).
- [5] Shor, P. W. Algorithms for quantum computation : discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 124–134 (1994).
- [6] Grover, L. K. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing*, STOC '96, 212–219 (ACM, New York, NY, USA, 1996). URL <http://doi.acm.org/10.1145/237814.237866>.
- [7] Bennett, C. H., Bessette, F., Brassard, G., Salvail, L. & Smolin, J. Experimental quantum cryptography. *Journal of Cryptology* **5**, 3–28 (1992). URL <https://doi.org/10.1007/BF00191318>.
- [8] Lanyon, B. *et al.* Towards quantum chemistry on a quantum computer. *Nature chemistry* **2**, 106–111 (2010).
- [9] Resch, S. & Karpuzcu, U. R. Quantum computing : An overview across the system stack (2019). [1905.07240](https://arxiv.org/abs/1905.07240).
- [10] Loss, D. & DiVincenzo, D. P. Quantum computation with quantum dots. *Physical Review A* **57**, 120–126 (1998). URL <http://dx.doi.org/10.1103/PhysRevA.57.120>.
- [11] DiVincenzo, D. P. The physical implementation of quantum computation. *Fortschritte der Physik* **48**, 771–783 (2000). URL [http://dx.doi.org/10.1002/1521-3978\(200009\)48:9/11<771::AID-PROP771>3.0.CO;2-E](http://dx.doi.org/10.1002/1521-3978(200009)48:9/11<771::AID-PROP771>3.0.CO;2-E).

- [12] Kane, B. Silicon-based quantum computation. *Fortschritte der Physik* **48**, 1023–1041 (2000). URL [http://dx.doi.org/10.1002/1521-3978\(200009\)48:9/11<1023::AID-PROP1023>3.0.CO;2-J](http://dx.doi.org/10.1002/1521-3978(200009)48:9/11<1023::AID-PROP1023>3.0.CO;2-J).
- [13] Saeedi, K. *et al.* Room-temperature quantum bit storage exceeding 39 minutes using ionized donors in silicon-28. *Science* **342**, 830–833 (2013). URL <https://science.sciencemag.org/content/342/6160/830>. <https://science.sciencemag.org/content/342/6160/830.full.pdf>.
- [14] Yoneda, J. *et al.* A quantum-dot spin qubit with coherence limited by charge noise and fidelity higher than 99.9%. *Nature Nanotechnology* **13**, 102–106 (2017). URL <http://dx.doi.org/10.1038/s41565-017-0014-x>.
- [15] Vandersypen, L. M. K. *et al.* Interfacing spin qubits in quantum dots and donors—hot, dense, and coherent. *npj Quantum Information* **3** (2017). URL <http://dx.doi.org/10.1038/s41534-017-0038-y>.
- [16] Zajac, D. M., Hazard, T. M., Mi, X., Nielsen, E. & Petta, J. R. Scalable gate architecture for a one-dimensional array of semiconductor spin qubits. *Phys. Rev. Applied* **6**, 054013 (2016). URL <https://link.aps.org/doi/10.1103/PhysRevApplied.6.054013>.
- [17] Botzem, T. *et al.* Tuning methods for semiconductor spin qubits. *Physical Review Applied* **10** (2018). URL <http://dx.doi.org/10.1103/PhysRevApplied.10.054026>.
- [18] van der Wiel, W. G. *et al.* Electron transport through double quantum dots. *Rev. Mod. Phys.* **75**, 1–22 (2002). URL <https://link.aps.org/doi/10.1103/RevModPhys.75.1>.
- [19] Ando, T., Fowler, A. B. & Stern, F. Electronic properties of two-dimensional systems. *Rev. Mod. Phys.* **54**, 437–672 (1982). URL <https://link.aps.org/doi/10.1103/RevModPhys.54.437>.
- [20] van Wees, B. J. *et al.* Quantized conductance of point contacts in a two-dimensional electron gas. *Phys. Rev. Lett.* **60**, 848–850 (1988). URL <https://link.aps.org/doi/10.1103/PhysRevLett.60.848>.
- [21] Ishikuro, H. & Hiramoto, T. Quantum mechanical effects in the silicon quantum dot in a single-electron transistor. *Applied Physics Letters* **71**, 3691–3693 (1997). URL <https://doi.org/10.1063/1.120483>. <https://doi.org/10.1063/1.120483>.
- [22] Baart, T. A., Eendebak, P. T., Reichl, C., Wegscheider, W. & Vandersypen, L. M. K. Computer-automated tuning of semiconductor double quantum dots into the single-electron regime. *Applied Physics Letters* **108**, 213104 (2016). URL <http://dx.doi.org/10.1063/1.4952624>.
- [23] Duda, R. O. & Hart, P. E. Use of the hough transformation to detect lines and curves in pictures. *Commun. ACM* **15**, 11–15 (1972). URL <http://doi.acm.org/10.1145/361237.361242>.

- [24] Fernandes, L. A. F. & Oliveira, M. M. Real-time line detection through an improved hough transform voting scheme. *Pattern Recogn.* **41**, 299–314 (2008). URL <http://dx.doi.org/10.1016/j.patcog.2007.04.003>.
- [25] Gioi, R., Jakubowicz, J., Morel, J.-M. & Randall, G. Lsd : A fast line segment detector with a false detection control. *IEEE transactions on pattern analysis and machine intelligence* **32**, 722–32 (2010).
- [26] Lapointe-Major, M. *et al.* Algorithm for automated tuning of a quantum dot into the single-electron regime (2019). [arXiv:1911.11651](https://arxiv.org/abs/1911.11651).
- [27] Zwolak, J. P., Kalantre, S. S., Wu, X., Ragole, S. & Taylor, J. M. Qflow lite dataset : A machine-learning approach to the charge states in quantum dot experiments. *PLOS ONE* **13**, e0205844 (2018). URL <http://dx.doi.org/10.1371/journal.pone.0205844>.
- [28] Mills, A. R. *et al.* Computer-automated tuning procedures for semiconductor quantum dot arrays. *Applied Physics Letters* **115**, 113501 (2019). URL <https://doi.org/10.1063/1.5121444>. <https://doi.org/10.1063/1.5121444>.
- [29] Griffiths, D. J. *Introduction to electrodynamics; 4th ed.* (Pearson, Boston, MA, 2013). URL <https://cds.cern.ch/record/1492149>. Re-published by Cambridge University Press in 2017.
- [30] Kalantre, S. S. *et al.* Machine learning techniques for state recognition and auto-tuning in quantum dots. *npj Quantum Information* **5** (2019). URL <http://dx.doi.org/10.1038/s41534-018-0118-7>.
- [31] MATLAB. *R2017a* (The MathWorks Inc., Natick, Massachusetts, 2017). URL <https://www.mathworks.com/>.
- [32] Thompson, C. & Shure, L. *Image Processing Toolbox: For Use with MATLAB;[user's Guide]* (MathWorks, 1995).
- [33] Notebookcheck. *Dell XPS 17-L702X review* (2011). URL <https://www.notebookcheck.net/Dell-XPS-17-L702X.50162.0.html>.
- [34] Wong, S. C., Gatt, A., Stamatescu, V. & McDonnell, M. D. Understanding data augmentation for classification : when to warp ? (2016). [1609.08764](https://arxiv.org/abs/1609.08764).
- [35] Weiss, G. Special conics in a hyperbolic plane (2017).
- [36] Griffiths, D. J. & Schroeter, D. F. *Introduction to Quantum Mechanics* (Cambridge University Press, 2018), 3 edn.
- [37] Holzmann, G. *Beyond Photography: The Digital Darkroom* (Prentice Hall Professional Technical Reference, 1988).
- [38] Eddins, S. Spatial transformations : Defining and applying custom transforms. <https://blogs.mathworks.com/steve/2006/08/04/>

- [spatial-transformations-defining-and-applying-custom-transforms/](#) (2006).
Accessed : 2019-10-08.
- [39] Zhu, J.-Y., Park, T., Isola, P. & Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks (2017). [1703.10593](#).
- [40] Isola, P., Zhu, J.-Y., Zhou, T. & Efros, A. A. Image-to-image translation with conditional adversarial networks (2016). [1611.07004](#).
- [41] Yuheng, S. & Hao, Y. Image segmentation algorithms overview (2017). [1707.02051](#).
- [42] Cireşan, D. C., Meier, U., Gambardella, L. M. & Schmidhuber, J. Deep, big, simple neural nets for handwritten digit recognition. *Neural Computation* **22**, 3207–3220 (2010). URL http://dx.doi.org/10.1162/NECO_a_00052.
- [43] Le, Q. V. *et al.* Building high-level features using large scale unsupervised learning (2011). [1112.6209](#).
- [44] Helbing, D. Societal, economic, ethical and legal challenges of the digital revolution : From big data to deep learning, artificial intelligence, and manipulative technologies (2015). [1504.03751](#).
- [45] Forsyth, D. A. & Ponce, J. *Computer Vision: A Modern Approach* (Prentice Hall Professional Technical Reference, 2002).
- [46] Taigman, Y., Polyak, A. & Wolf, L. Unsupervised cross-domain image generation (2016). [1611.02200](#).
- [47] Konno, T. *et al.* Deep learning model for finding new superconductors (2018). [1812.01995](#).
- [48] Isayev, O. *et al.* Materials cartography : Representing and mining materials space using structural and electronic fingerprints. *Chemistry of Materials* **27**, 735–743 (2015). URL <https://doi.org/10.1021/cm503507h>. <https://doi.org/10.1021/cm503507h>.
- [49] Mansouri Tehrani, A. *et al.* Machine learning directed search for ultraincompressible, superhard materials. *Journal of the American Chemical Society* **140**, 9844–9853 (2018). URL <https://doi.org/10.1021/jacs.8b02717>. PMID : 30010335, <https://doi.org/10.1021/jacs.8b02717>.
- [50] Stanev, V. *et al.* Machine learning modeling of superconducting critical temperature. *npj Computational Materials* **4** (2018). URL <http://dx.doi.org/10.1038/s41524-018-0085-8>.