Brain-inspired computing leveraging the transient non-linear dynamics of magnetic nano-oscillators

Mathieu Riou

NNT : 2019SACLS033

Thèse de doctorat

Brain-inspired computing leveraging the transient non-linear dynamics of magnetic nano-oscillators

Thèse de doctorat de l'Université Paris-Saclay
préparée à l'Université Paris-Sud

Ecole doctorale n°564 Physique en Ile-de-France (EDPIF)
Spécialité de doctorat : Physique

Thèse présentée et soutenue à Palaiseau, le 23/01/2019, par

MATHIEU RIOU

Composition du Jury :

Laurent Larger Professeur, Université de Franche-Comté (FEMTO-ST)	Président, Rapporteur
Karin Everschor-Sitte Research group leader, University of Mainz	Rapportrice
Laura Thevenard Chargée de recherche, Université Pierre et Marie Curie (CR INSP)	Examinatrice
François Duport Ingénieur-chercheur, III-V lab	Examinateur
Damien Querlioz Chargé de recherche, Université Paris Saclay (C2N)	Examinateur
Julie Grollier Directrice de recherche, Unité mixte de Physique CNRS/Thales	Directrice de thèse

CONTENTS

REMERCIEMENTS

Cette thèse est le fruit de trois années exceptionnelles que j'ai passé au sein de l'unité Mixte de Physique CNRS/Thales.

Tout d'abord je tenais à remercier les membres du jury Laurent Larger, Karin Everschor-Sitte, Laura Thévenard, François Duport et Damien Querlioz. Merci pour le temps que vous avez passé à la lecture et la correction de mon manuscrit. Merci à Laurent Larger et Karin Everschor-Sitte d'avoir été rapporteurs de cette thèse, merci à Laura Thévenard, François Duport et Damien Querlioz d'avoir examiné la soutenance.

Si pour certains l'expériences du doctorat peut-être éprouvante, il n'en a été rien pour moi et cela est en grande partie dû à l'atmosphère particulière de ce laboratoire. Je souhaite donc remercier Frédéric Pettroff et Frédéric Nguyen Van Dau qui par leur direction ont su instaurer ce climat, ainsi que l'ensemble du personnel permanents et non permanent pour leur bienveillance.

En tout premier lieu je souhaite remercier du fond du cœur Julie Grollier qui m'a donné l'opportunité de réaliser cette thèse et qui par bien des aspects a été une encadrante exceptionnelle. Tout d'abord ce que je sais de la spintronique et des approches neuromorphiques je te le dois essentiellement. J'ai toujours été et je ne cesserais d'être impressionné par l'étendue de tes connaissances et ta maîtrise sur des sujets extrêmements divers, ainsi que par ton incroyable créativité. À chaque problème que je voyais tu semblais toujours avoir dix possibles solutions. Mais outre tes qualités scientifiques exceptionnelles, c'est tout d'abord pour tes qualités humaines que je voudrais te remercier. Alors que tu as été bien souvent débordée de travail et de responsabilités, tu as toujours trouvé du temps pour tes doctorants, pour nous écouter et nous conseiller. Tu as permis que la thèse se passe dans les meilleures conditions et aussi tu m'as beaucoup aidé pour l'après thèse.

Je souhaite aussi remercier Jacob Torrejon et Flavio Abreu Araujo, post-docs dans l'équipe qui ont été mes "sensei" et les piliers de la "fine équipe". Jacob tu m'as transmis ta passion pour l'expérience et ce

avec beaucoup de bienveillance. Je retiendrai toujours ce que tu as dit la première fois que j'ai mené une expérience qui correspond aussi à la première fois où j'ai détruit du matériel de laboratoire: "Ne t'inquiète pas, en expérience celui qui ne casse rien c'est celui qui ne fait rien". Pour moi tu as toujours été un artiste qui en quelques mouvements, pouvait faire fonctionner une expérience sur laquelle je m'acharnais depuis des heures. Flavio quant à toi tu m'as appris à coder et ce n'était pas tâche aisée vu comment j'ai pu faire du "vilain" dans mes programmes. Je gardais en mémoire évidemment tes précieux conseils de programmation mais aussi nos nombreuses escapades nocturnes. Enfin en plus de m'avoir encadré et appris à programmer, tu étais aussi un ami toujours là pour remonter le moral dans les moments de stress ou de creux qui sont inhérents à la recherche.

Je voudrais remercier Philippe Talatchian qui est littéralement mon "frère de thèse", pour le soutien mutuel durant le stage et le doctorat (et tout particulièrement pour la rédaction de thèse où on s'est serré les coudes). Tu étais toujours là pour aider ou remonter le moral, outre le fait que tu étais toujours prêt à prendre un verre dans ton noble rôle de mousquetaire.

Je remercie aussi Mafalda et Salvo qui ont eu l'abnégation de relire une partie de mon manuscrit je vous dis donc "obrigado Mafalda" et "grazie Salvo".

Je dis merci à Nathan que j'ai eu la chance d'encadrer et je te souhaite te t'amuser autant que moi durant ta thèse. Je te souhaite à toi ainsi qu'à Danijela, Alice, Anastasia et Maxence plein de succès pour la suite du neuromorphique.

Je voudrais remercier Sumito Tsunegi pour ses précieux échantillons, quasi parfaits et Tomohiro Taniguchi pour ses précieux conseils sur le Japon.

Je remercie aussi Mark Stiles, Damien Querlioz, Vincent Cros et Paolo Bortolotti pour leurs précieux conseils et corrections notamment pour la rédaction d'articles.

Je remercie Bogdan pour les discussions sur le reservoir computing, mais aussi pour les bons moments en conférence et ses délicieuses pizza.

Je remercie Frédéric Nguyen Van Dau, Juan Trastoy, Olivier D'Allivy Kelly, Paolo Bortolotti et Daniel Dolfi pour leur aide et leurs précieux conseils sur le fonctionnement de Thales.

Je remercie Anne Dussart, Nathalie Lesauvage et Brigitte Marchand qui notamment se sont toujours démenées pour gérer mes missions.

Je voudrais remercier aussi tous les post-docs et doctorants de l'unité qui ont permis de créer cette super ambiance et avec qui j'ai passé de bonne journées et soirées. Plus particulièrement, je remercie Eliana pour toutes les bonnes droites qu'elle m'a mise dans la tête à la boxe et tous les éclats de rire qu'elle a eu pendant ces presque quatres ans; Xavier, autre grand mousquetaire devant l'éternel avec qui on a partagé soirées au bar et séances de boxe; Samh avec qui j'ai passé de super moments à refaire le monde et à fumer des cigares, Maëlis avec qui je refaisais aussi le monde et qui était toujours la pour me conseiller; Steffen et son exceptionnel vin chaud; Maxence qui n'avait de cesse de me rappeler que "je ne payais pas de mine mais...", Seiha et son humour, Huong et sa gentillesse, Jason pour les soirées techno, Adrian pour son humour noir et ses conseils sur le machine learning, Sophie pour ses débats enflammés et ses histoires "de dingues".

Je remercie aussi mes collègues de la cité des sciences notamment Olivier, Alain, Nadège et Julia qui m'ont donné envie de rendre la science accessible à tous. Je remercie aussi les doctorants en mission à la cité des sciences avec qui j'ai partagé d'excellents moments en médiation et notamment Louis, Cyril, Joel, Justin, Victor, Hélène et Jan.

Je remercie mes amis et notamment ceux qui comme Kevish et Linh Thao ont su venir prendre l'apéro et squatter mon canapé dans les moments où j'avais besoin de compagnie. Je remercie Grégoire notamment pour les vacances de "bromance" durant ma thèse, Hakim pour le sarcasme et le body karate et Jean toujours prêt à me soutenir moralement un verre à la main. Je remercie Rénima, Lucas B., Lucas H., Letian, Madeleine et aussi ma cousine Marie qui ont bravé la neige pour venir assister à ma soutenance.

Je remercie Marie-Liesse. Merci pour ton soutien et ta présence qui a éclairé ma dernière année de thèse.

Enfin je remercie mes parents et ma grande sœur (mon éternel mentor) qui me soutiennent inconditionnellement depuis ma naissance (et il faut bien reconnaître que cela représente un temps conséquent).

SUMMARY

Title: Brain-inspired computing leveraging the transient non-linear dynamics of magnetic nano-oscillators

Keywords: Spintronics, Neural networks (computer science)

This thesis studies experimentally the transient dynamics of magnetic nano-oscillators for brain-inspired computing. For pattern recognition tasks such as speech or visual recognition, the brain is much more energy efficient than classical computers. Developing brain-inspired chips opens the path to overcome the limitations of present processors and to win several orders of magnitude in the energy consumption of data processing. The efficiency of the brain originates from its architecture particularly well adapted for pattern recognition. The building blocks of this architecture are the biological neurons, which can be seen as interacting non-linear oscillators generating spatial chain reactions of activations. Nevertheless, the brain has one hundred billion neurons and a brain-inspired chip would require extremely small dimension oscillators. The spin-transfer torque oscillators (STNO) have nanometric size, they are fast (nanosecond time-scales), highly non-linear and their spin-torque dependent response is easily tunable (for instance by applying an external magnetic field or a d.c. current). They work at room temperature, they have a low thermal noise and they are compatible with CMOS technologies. Because of these features, they are excellent candidates for building hardware neural networks, which are compatible with the standard computers. In this thesis, we used a single STNO to emulate the behavior of a whole neural network. In this time multiplexed approach, the oscillator emulates sequentially each neuron and a temporal chain reaction replace the spatial chain reaction of a biological neural network. In particular, we used the relaxation and the non-linear dependence of the oscillation amplitude with the applied current to perform neuromorphic computing. One of the main results of this thesis is the demonstration of speech recognition (digits said by different speakers) with a state-

of-the-art recognition rate of 99.6%. We show that the recognition performance is highly dependent on the physical properties of the STNO, such as the linewidth, the emission power or the frequency. We thus optimized the experimental bias conditions (external applied magnetic field, d.c. current and rate of the input) in order to leverage adequately the physical properties of the STNO for recognition. Voice waveforms require a time-to-frequency transformation before being processed, and this step is performed numerically before the experiment. We studied the influence of different time-to-frequency transformations on the final recognition rate, shading light on the critical role of their non-linear behavior. Finally, in order to solve problems requiring memory, such as temporal sequence analysis, we measured the intrinsic memory of a STNO, which comes from the relaxation of the oscillation amplitude. We also increased this memory, using a delayed feedback loop. This feedback improved the range of memory from a few hundreds of nanoseconds to more than ten microseconds. This feedback memory allows suppressing up to 99% of the errors on a temporal pattern recognition task (discrimination of sine and square waveforms).

RÉSUMÉ

Titre: Calcul bio-inspiré utilisant la dynamique non-linéaire transitoire d'oscillateurs magnétiques nanométriques

Mots clés: Électronique de spin, Réseaux neuronaux (informatique)

L'objectif de cette thèse est la réalisation expérimentale de calcul bio-inspiré en utilisant la dynamique transitoire d'oscillateurs magnétique nanométriques. Pour bien des tâches telle que la reconnaissance vocale, le cerveau fonctionne bien plus efficacement en terme d'énergie qu'un ordinateur classique. Le développement de puces neuro-inspirées offre donc la perspective de surmonter les limitations des processeurs actuels et de gagner plusieurs ordres de grandeurs sur la consommation énergétique du traitement de données. L'efficacité du cerveau à traiter des données est due à son architecture, qui est particulièrement adaptée à la reconnaissance de motifs. Les briques de base de cette architecture sont les neurones biologiques. Ceux-ci peuvent être vus comme des oscillateurs non linéaires qui interagissent et génèrent des cascades spatiales d'activations en réponse à une excitation. Cependant le cerveau comprend cent milliards de neurones et le développement d'une puce neuro-inspiré requerrait des oscillateurs de très petite dimension. Les oscillateurs à transfert de spin (STNO) sont de taille nanométrique, ont une réponse rapide (de l'ordre de la nanoseconde), sont fortement non-linéaires et leur réponse dépendante du couple de transfert de spin est aisément ajustable (par exemple par l'application d'un courant continu ou d'un champ magnétique). Ils fonctionnent à température ambiante, ont un très faible bruit thermique, et sont compatible avec les technologies CMOS. Ces caractéristiques en font d'excellents candidats pour la réalisation de réseaux artificiels de neurones compatibles avec un ordinateur classique. Dans cette thèse, nous avons utilisé un unique STNO pour générer le comportement d'un réseau de neurones. Ainsi l'oscillateur joue à tour de rôle chaque neurone. Une cascade temporelle remplace donc la cascade spatiale

d'un réseau de neurones biologiques. En particulier nous avons utilisé la relaxation et la dépendance non-linéaire de l'amplitude des oscillations afin de réaliser du calcul neuromorphique. L'un des résultats principaux de cette thèse est la réalisation de reconnaissance vocale (reconnaissance de chiffres dits par 5 locuteurs différents) en obtenant un taux de reconnaissance à l'état de l'art de 99.6%. Nous avons pu montrer que les performances de la reconnaissance sont étroitement dépendantes des propriétés physiques du STNO tel que l'évolution de la largeur de raie, la puissance d'émission, ou la fréquence d'émission. Nous avons donc optimisé les conditions expérimentales (champs magnétiques et courant continu appliqués, fréquence du signal à traiter) afin de pouvoir utiliser au mieux les propriétés physiques du STNO pour la reconnaissance. Les signaux vocaux requièrent d'être transformés du domaine temporel au domaine fréquentiel, avant de pouvoir être traités, et cette étape est réalisée numériquement en amont de l'expérience. Nous avons étudié l'influence de différents prétraitements sur la reconnaissance et mis en évidence le rôle majeur de la non-linéarité de ces derniers. Enfin, afin de pouvoir traiter des problèmes requérant de la mémoire, tel que par exemple des signaux sous forme de séquences temporelles, nous avons mesuré la mémoire que possède intrinsèquement un STNO, du fait de sa relaxation. Nous avons aussi augmenté cette mémoire à l'aide d'une boucle à retard. Ce dispositif a permis d'accroître la plage de mémoire de quelques centaines de nanosecondes à plus d'une dizaine de microsecondes. L'ajout de cette mémoire extrinsèque a permis de supprimer jusqu'à 99% des erreurs sur une tâche de reconnaissance de motifs temporels (reconnaissance de signaux sinusoïdaux et carrés).

SYNTHÈSE EN FRANÇAIS (SUMMARY IN FRENCH)

INTRODUCTION

Récemment, l'intelligence artificielle s'est rapidement développée car ces algorithmes offrent la possibilité pour des ordinateurs de dépasser les performances humaines dans des tâches cognitives telles que la reconnaissance d'image ou la reconnaissance vocale. À l'ère de la donnée, l'intelligence artificielle devient indispensable dans de nombreux secteurs industriels afin d'analyser automatiquement des situations ambigües. Les algorithmes à la base de ces progrès sont les réseaux de neurones artificiels qui sont inspirés de la non-linéarité et de la plasticité des réseaux de neurones biologiques. Ils apparaissaient dans les années 50, où le premier algorithme permettant à une machine de d'apprendre des représentations abstraites fut développé [1]. Cependant ce n'est que récemment que ces algorithmes connurent un essor grâce notamment à l'amélioration des microprocesseurs. En effet pour résoudre des tâches pratiques telles que de la reconnaissance d'image, il est souvent nécessaire de calculer la réponse de millions de neurones à l'aide de centaines de millions de paramètres. Même si ces algorithmes sont loin de la complexité du cerveau humain qui compte 100 milliards de neurones, leur exécution sur un ordinateur classique est extrêmement coûteuse en énergie.

Les ordinateurs actuels sont fondés sur l'approche de Von Neuman où l'unité de calcul est physiquement séparée de l'unité de mémoire et où les données doivent être déplacées séquentiellement d'une unité à l'autre au travers d'un bus commun. Ce type d'architecture n'est pas adapté pour émuler des réseaux massivement parallèles et si un algorithme de réseau de neurones permet d'atteindre des résultats similaires aux performances humaines pour quelques tâches particulières, l'énergie dépensée lors de son exécution sur un ordinateur est trois à quatre ordres de grandeurs supérieure à la consommation énergétique du cerveau humain. L'efficacité énergétique du cerveau est due à son architecture massivement parallèle, où les neurones qui trait-

ent l'information sont densément interconnectés par des synapses qui portent en elles de la mémoire. Cette constatation motive le développement de puces inspirées du cerveau, intégrant des composants dont la physique imite le comportement des neurones. Les neurones biologiques encodent l'information dans les impulsions électriques qu'ils génèrent. Tout un pan des neurosciences et de la physique les modèle comme des oscillateurs non-linéaires. Dès lors, le cerveau peut être vu comme une assemblée d'oscillateurs non-linéaires qui calculent grâce à leur dynamique complexe. Une puce inspirée du cerveau basée sur ce principe devrait intégrer des millions d'oscillateurs non-linéaires sur une surface aussi faible qu'un centimètre carré ce qui requière des composants de taille nanométrique. Ainsi des nano-oscillateurs non-linéaires sont nécessaires pour construire des neurones matériels.

Malgré de nombreuses propositions incluant des neurones memrisitifs et des neurones à base de jonctions Josephson, il n'y a pas eu jusqu'à présent de démonstration de calcul neuromorphique utilisant un neurone artificiel nanométrique. En effet les composants nanométriques sont généralement bruités et peu fiables, ce qui limite leur emploi pour le calcul. Dans cette thèse nous utilisons la dynamique non-linéaire d'un nano-oscillateur à transfert de spin car malgré leur taille nanométrique, ces composants magnétiques ont un long cycle de vie et un fort rapport signal sur bruit.

Cette thèse est la première démonstration expérimentale de calcul neuromorphique utilisant un neurone nanométrique. Pour cette démonstration, nous avons utilisé la dynamique non-linéaire transitoire d'un oscillateur à transfert de spin et nous avons obtenu des résultats à l'état de l'art pour une tâche de reconnaissance de chiffres parlés [2].

CALCUL NEUROMORPHIQUE

Le cerveau humain peut associer très rapidement des situations similaires mais non identiques tel qu'une personne sur différentes photos.

Les capacités du cerveau sont dues à son architecture qui est encore un vaste sujet de recherche. Le cerveau est composé de neurones interconnectés par des synapses. Les synapses en changeant la connexion entre deux neurones permettent l'apprentissage. Les neurones, qui

sont le centre d'intérêt de cette thèse sont des auto-oscillateurs qui possèdent mémoire et non-linéarité.

Les algorithmes de réseaux de neurones artificiels s'inspirent de la non-linéarité des neurones biologiques et de la plasticité des synapses. Ces algorithmes appartiennent à la classe de l'apprentissage supervisé où durant une phase d'apprentissage les paramètres de l'algorithme sont réglés (ici il s'agit des pondérations des synapses) et d'une phase d'inférence où les paramètres restent fixe.

Exécuter ces algorithmes sur un ordinateur classique est énergivore, car les données font de nombreux allers-retours entre la mémoire et le processeur. Une perspective de recherche prometteuse est donc d'implémenter des puces dont l'architecture s'inspire elle aussi du cerveau.

État de l'art des neurones matériels pour le calcul neuromorphique

Le CMOS est une technologie établie ce qui explique que les systèmes neuromorphiques les plus avancés ont été développés avec cette approche. Mais émuler les fonctionnalités d'un neurone requière des circuits de transistors complexes qui occupent une surface trop importante pour créer la densité de neurones nécessaire à une puce neuromorphique.

Il y a donc un besoin de nouveaux composants nanométriques pour émuler les propriétés d'un neurone. La première approche présentée est le neurone memristif. Même si les memristors peuvent être de taille nanométrique, ils doivent être intégrés avec d'autres composants de plus grande taille (notamment des condensateurs) pour émuler un neurone. De plus, la principale faiblesse des memristors à l'heure actuelle est leur cycle de vie relativement réduit. Enfin, les travaux pour résoudre des tâches de reconnaissance avec de tels composants sont encore préliminaires. Des neurones à base de jonctions Josephson ont aussi été proposés. Ces composants ont la possibilité d'être nanométriques, même si la démonstration actuelle est bien plus grande. Ils fonctionnent rapidement et à faible énergie. En revanche ils nécessitent une température basse pour fonctionner et aucune tâche de reconnaissance n'a été démontrée. Finalement, les oscillateurs électromécaniques. Ces oscillateurs électromécaniques fonctionnent à

basse énergie et peuvent être couplés. Cette approche est très récente (2017). Des premiers résultats de reconnaissance de motifs ont été obtenues en 2018 mais avec un dispositif micrométrique.

Phénomènes physiques impliqués dans le fonctionnement des oscillateurs à transfert de spin

Les oscillateurs à transfert de spin utilisent pour leur fonctionnement une association des phénomènes de magnétorésistance et de couple de transfert de spin. Dans un matériau ferromagnétique, la diffusion des électrons dépend de leur spin par rapport à l'aimantation du matériaux. Il en résulte que la résistance d'un empilement composée de deux couches ferromagnétiques séparées par un matériau non magnétique dépend de l'angle relatif entre les deux aimantations des couches magnétiques.

À l'inverse, lorsqu'une couche ferromagnétique reçoit un courant polarisé en spin, ce dernier applique un couple sur l'aimantation de la couche ferromagnétique appelé couple de transfert de spin. Pour des couples suffisamment importants, l'aimantation est déstabilisé et peu se renverser ou précesser autour de sa position d'équilibre.

Principe des oscillateurs à transfert de spin et avantage pour le neuromorphique

Un oscillateur à transfert de spin est constitué de deux couches ferromagnétiques séparées par une couche non-magnétique. Lorsqu'un courant continu suffisant est injecté, l'aimantation de la couche supérieure, appelée couche libre, précesse autour de sa position d'équillibre. Il en résulte l'émission d'une tension oscillante. Ces oscillateurs qui émergent au début des années 2000 ont été optimisés au cours des années et possèdent désormais un comportement stable et peu bruité. Ces composants sont intrinsèquement non-linéaires et sont soumis à un phénomène de relaxation puisque l'aimantation de la couche libre nécessite un temps fini pour changer sa trajectoire. Ils ont été initialement développés pour servir d'émetteur et de détecteur de signaux radio-fréquence. Cependant en temps qu'auto-oscillateurs non-linéaires, ils pourraient émuler le comportement d'un neurone.

"reservoir computing" avec un unique composant non-linéaire

Le "reservoir computing" est un type de réseau de neurones artificiel. Une entrée est injectée dans un réseau de neurones récurent dont les connexions sont fixes et arbitraires. Ce réseau est appelé réservoir. Le réservoir projette non-linéairement l'entrée initiale dans un espace de plus grande dimension où chaque nouvelle coordonnée est donnée par l'état d'un neurone du réservoir. Si le réservoir possède les propriétés d'approximation, de séparation et de mémoire requise, une fois projeté en plus grande dimension, le problème peut être résolu linéairement. Cette projection non-linéaire peut être obtenue en utilisant un unique composant non-linéaire ce qui simplifie les implémentations expérimentales. L'entrée doit être multiplexée temporellement, et l'unique composant non-linéaire joue le rôle de plusieurs neurones temporels. En connaissant le multiplexage temporel, la projection non-linéaire du réservoir peut être extraite de la réponse temporelle de l'unique composant non-linéaire. Cette approche a été utilisée notamment en optique où des tâches de reconnaissance tel que la reconnaissance de mots ont été éffectuées. Nous utiliserons cette approche par la suite pour réaliser des tâches de reconnaissance avec un oscillateur à transfert de spin.

implémentation spintronique de "reservoir computing"

Le composant que nous utilisons est un oscillateur à transfert de spin dont la couche libre est un vortex et dont la couche non-magnétique est un isolant. Cet oscillateur a été choisi pour sa stabilité et sa forte puissance. Dans le circuit expérimental, l'oscillateur est connecté à une source arbitraire de tension qui envoie l'entrée multiplexée temporellement. Un électroaimant et une source de courant continu fixent le point de fonctionnement de l'oscillateur. La réponse temporelle de l'oscillateur est mesurée grâce à un oscilloscope. Réaliser du reservoir computing requière de la non-linéarité et de la mémoire. La dépendance de l'amplitude des oscillations de tension émises avec le courant d'entrée sert de non linéarité. La relaxation de l'amplitude des oscillations fait office de mémoire.

RÉSULTATS DE CLASSIFICATION

Deux tâches ont été utilisées pour évaluer les performances du reservoir computing spintronique: la classification de signaux sinusoïdaux et carrés ainsi que la reconnaissance de chiffres parlés. La classification de signaux sinusoïdaux et carrés est une tâche plus simple mais qui permet de tester la non-linéarité et la mémoire du réservoir. En utilisant 24 neurones temporels, nous avons pu parfaitement reconnaître les signaux sinusoïdaux et carrés. Cependant ces résultats sont obtenus dans des conditions optimales de point de fonctionnement.

La reconnaissance de chiffres parlés requière une étape de transformation acoustique avant que l'entrée ne soit multiplexée temporellement. Nous avons utilisé deux méthodes qui sont la décomposition cochléaire et un spectrogramme. Dans le cas du spectrogramme, l'utilisation de l'oscillateur ajoute 70% au taux de reconnaissance final qui est de de 80%, ce qui montre bien le rôle crucial de l'oscillateur. En utilisant la décomposition cochléaire comme transformation acoustique, nous obtenons un taux de reconnaissance de 99.6% ce qui est à l'état de l'art à la fois pour les approches logicielles et matérielles. Cependant la décomposition cochléaire sépare déjà de façon importante les entrée et utilisée seule, elle permet déjà de reconnaître correctement 96% des chiffres parlés.

OPTIMISATION DES PARAMÈTRES EXPÉRIMENTAUX ET TRAITEMENT DE DONNÉES POUR L'AMÉLIORATION DE LA CLASSIFICATION

Le temps d'échantillonnage de l'entrée multiplexée temporellement θ doit être suffisamment court pour assurer à la fois la mémoire du réservoir et la connexion entre neurones temporelles sans pour autant dégrader de façon trop importante le rapport signal sur bruit. Le meilleur compromis trouvé est un temps égal à la moitié du temps de relaxation de l'oscillateur ($\theta = T_{relax}/2$). En ajoutant numériquement un décalage temporel, on supprime le besoin de mémoire et les résultats dépendent essentiellement de la non-linéarité et du rapport signal sur bruit. Le point de fonctionnement joue un rôle important pour optimiser ces propriétés. Tout d'abord, le champ magnétique appliqué doit avoir une valeur intermédiaire (entre 300 mT et 500 mT)

pour que la variation de l'amplitude des oscillations ne soit ni trop abrupte ni trop douce. Le courant continu quant à lui doit être choisi pour obtenir de grandes variations d'amplitude des oscillations. Plus généralement, le point de fonctionnement doit être choisi dans un régime intermédiaire de bruit et de non-linéarité.

LE RÔLE DE LA NON-LINÉARITÉ DANS LES TRANSFORMATIONS ACOUSTIQUES POUR LA RECONNAISSANCE VOCALE

Différentes transformations acoustiques, à savoir le MFCC, la décomposition cochléaire et le spectrogramme, ont été étudiées comme descripteur de l'information. En apliquant une régression linéaire et une fonction maximum, le MFCC et la décomposition cochléaire donnent des taux de reconnaissance de chiffre parlé élevées, à l'inverse du spectrogramme qui ne sépare pas les différentes classes de mot. En ajoutant de la non-linéarité au spectrogramme avec un exposant puissance, nous avons montré que l'on pouvait obtenir des taux de reconnaissance aussi élevés que pour les deux autres méthodes, ce qui souligne que c'est le caractère non-linéaire de la transformation qui permet la reconnaissance mais que le type de non-linéarité importe dans une bien moindre mesure.

Dans un second temps, nous avons évalué quel gain de taux de reconnaissance conférait l'utilisation d'un oscillateur à transfert de spin. Ces résultats sont à la fois le fruit de simulations et d'expériences. Pour le MFCC et la décomposition cochléaire, le gain est modéré malgré un taux de reconnaissance proche de 100% car la transformation acoustique seule permet déjà un taux de reconnaissance élevé, ne laissant que peu de marge d'amélioration. En revanche pour le spectrogramme le gain particulièrement élevé même si le taux de reconnaissance final est aux alentours de 70-80%. Le taux de reconnaissance finale doit donc être interprété avec précaution selon la transformation acoustique choisie.

Implémentation expérimentale et preuve d'une mémoire à court terme

Nous avons dans un premier temps mesuré le temps de relaxation de l'oscillateur qui donne une évaluation de la plage de sa mémoire intrinsèque. Ce temps a été mesuré entre 100 ns et 300 ns selon le point de fonctionnement. Dans un second temps la plage de mémoire a été augmentée par l'ajout d'une boucle de rétroaction dans le circuit. Quand l'oscillateur reçoit un signal d'entrée, de multiples échos de cette perturbation sont observés sur des plages de temps jusqu'à 69 μs. Finalement la capacité de mémoire de l'oscillateur avec boucle de rétroaction a été évaluée à environ 3, ce qui signifie que des entrées jusqu'à trois pas de temps dans le passé peuvent être retrouvées.

Amélioration de la séparation des données pour des cas requérant de la mémoire

La classification de signaux sinusoïdaux et carrés permet de tester la mémoire dans un contexte où de la non-linéarité est aussi requise. Le pas de temps θ a été volontairement choisi plus grand afin d'effacer les effets de mémoire dûs à la relaxation. L'ajout de la boucle de rétroaction modifie les traces temporelles de l'amplitude d'oscillation. En particulier des valeurs d'entrée identiques dans les sinus et carré génèrent des traces temporelles différentes. Finalement nous avons créé une méthode de visualisation des données qui permet de voir à la fois la séparation entre sinus et carrés et les différents groupements naturels qui apparaissent dans les projections effectuées par le réservoir. Cette méthode confirme sans ambiguïté la présence d'une mémoire qui permet de séparer les différentes entrées. L'ajout d'une boucle de rétroaction réduit le taux d'erreur de 10.8% à 0.14%, ce qui prouve que la mémoire apportée par la boucle de rétroaction peut être utilisée pour le calcul.

INFLUENCE DU POINT DE FONCTIONNEMENT SUR L'EFFICACITÉ DE LA BOUCLE DE RÉTROACTION

La boucle de rétroaction a un effet différent selon le point de fonctionnement. Si dans la majorité des cas elle apporte de la mémoire, elle peut être aussi source de nouvelles erreurs qui peuvent compenser la plus value de la mémoire. En général la rétroaction est d'un moindre intérêt lorsque les erreurs initialement ne sont pas dues a un manque de mémoire, ce qui le cas pour les points de fonctionnement où le temps de relaxation est particulièrement élevé ainsi que ceux pour lequel le niveau de bruit est élevé. Au contraire la rétroaction est particulièrement bénéfique dans des régimes de fort rapport signal sur bruit et de faible temps de relaxation.

CONCLUSION

Cette thèse constitue la première démonstration de calcul neuromorphique utilisant un neurone matériel de taille nanométrique. Nous avons utilisé un unique composant dans l'approche du reservoir computing. Pour cette démonstration nous avons tiré parti de la non-linéarité de l'amplitude des oscillations et de la mémoire conférée par la relaxation de l'aimantation de la couche libre. Grace à cela nous avons classifié parfaitement des signaux sinusoïdaux et carrés et reconnu des chiffres dits par différentes personnes et ce avec un succès à l'état de l'art de 99.6%. Pour obtenir ces résultats, nous avons optimisé les paramètres expérimentaux tels que le pas de temps de l'entrée multiplexée temporellement et le point de fonctionnement de l'oscillateur. Nous avons aussi étudié en détail l'effet de différentes transformations acoustiques pour la reconnaissance des chiffres parlés. Finalement nous avons augmenté la plage de mémoire de l'oscillateur en ajoutant une boucle de rétroaction. Nous avons démontré l'efficacité de cette rétroaction en supprimant les erreurs dues à un manque de mémoire pour la classification de signaux sinusoïdaux et carrés.

Durant cette thèse nous avons utilisé un unique oscillateur avec une approche de multiplexage temporel, ce qui nous a permis de réaliser cette première démonstration. Pour des applications réelles, cette approche présente des limites, puisqu'elle contraint à pré-traiter l'entrée, elle ne permet que certaines architecture de réseaux de neurones et

pour de large réseaux de neurones le temps de réponse est trop long. Pour remédier à ces limitations, une perspective prometteuse est de construire de large réseaux d'oscillateurs à transfert de spin interconnectés. Ceci demande d'étudier le moyen de couplage le plus approprié. Pour réduire l'énergie que consommeraient de tels réseaux, on peut envisager l'usage d'oscillateurs à transfert de spin de taille plus réduite. Enfin pour bâtir une puce parfaitement autonome, celle-ci devrait être capable de réaliser de l'apprentissage, ce qui peut être obtenu par exemple en utilisant des connections réglables entre les oscillateurs. Dans cette optique, les memristors spintroniques pourraient réaliser de telles fonctions. Enfin les méthodes d'apprentissage utilisées dans les algorithmes de réseaux de neurones sont difficilement transposables aux architectures matérielles. Il serait donc intéressant d'étudier de nouvelles méthodes d'apprentissage tirant partie de la physique des composants.

INTRODUCTION

In recent years, artificial intelligence attracted much interest because using these algorithms offers the possibility for machines to outperform humans at cognitive tasks such as image recognition or speech recognition. In a data driven society, artificial intelligence should become more and more essential for many industries in order to analyze automatically ambiguous situations. The algorithms at the base of this progress are artificial neural networks, which take inspiration from the plasticity and non-linearity of biological neural networks. They originate in the fifties, where the first algorithm allowing a machine to learn abstract representations was developed [1]. However it is only recently that artificial neural networks were popularized, because the improvement of microprocessor computation capabilities enabled to run these complex algorithms. Indeed, solving useful tasks such as image recognition requires computing the response of millions of formal neurons and tuning tens of millions of parameters. Even though these algorithms are still far from the complexity of the human brain, which has one hundred billion neurons, running them on classical computer architectures is already energy costly.

Indeed the computers we use are based on the Von Neuman architecture where the processing unit is separated from the memory and data has to move sequentially back and forth between these two units through a shared bus. This particular architecture is not well adapted to emulate densely parallel networks, and if running an artificial neural network algorithm can lead to the same performance as humans for some particular tasks, the energy spent by a classical computer architecture is three to four orders of magnitude larger than the energy consumption of the human brain. The energy efficiency of the brain comes from its massively parallel architecture, where biological neurons which process the information are densely interconnected by synapses which hold the memory. This observation motivated building brain-inspired chips, with analog components whose physics mimics the behavior of the neurons. Biological neurons encode their information in the spikes they emit. A whole branch of neuroscience

and physics models them as non-linear auto-oscillators and the brain can be seen as large assembly of interconnected non-linear oscillators which computes through its complex dynamics. A brain-inspired chip based on these principles would thus require integrating millions of non-linear oscillators in an area as small as one centimeter square, which requires nanoscale devices. Therefore non-linear nano-oscillator components are needed to build such hardware neurons.

Beside multiple proposals including memristive neurons and Josephson junction neurons, there was no experimental demonstration of neuromorphic computing using a nanoscale neuron. Indeed nanoscale components generally tend to be noisy and unreliable, limiting their use for computation. In this thesis the non-linear behavior of a spin-torque nano-oscillator is leveraged because beside their nanometric size, these magnetic components have a long lifetime and a high signal to noise ratio.

This thesis presents the first experimental demonstration of neuromorphic computing with a nanoscale neuron. For this demonstration the transient non-linear dynamics of a spin-torque oscillator was used, and spoken digit recognition was achieved with state of the art results [2]. The following manuscript is organized in five parts.

In part i, we motivate the need for hardware brain-inspired computing. In chapter 1, we highlight that the computational power of the brain is due to its architecture. It is shown that neurons can be seen as non-linear oscillators. The software computing methods taking inspiration of the brain reach high performance but running them on classical computers is energy costly, which motivates the need for computers with a brain-inspired hardware architecture. In chapter 2, we show that the CMOS based neuromorphic approaches encounter scalability issues, and despite multiple proposal using alternative nano components, brain-inspired computing with a nanoscale hardware neuron remains yet to demonstrated.

In part ii, spin-torque oscillators which are the non-linear oscillators used in this thesis are presented. In chapter 3, the magnetoresistance and spin-torque phenomena which are at the basis of spin-torque oscillators are presented. In chapter 4, we show how by leveraging both phenomena, spin-torque oscillators emit oscillating voltages. The increase of their performance in terms of power emission and noise made them suitable for practical applications. Using the different

theoretical models predicting their dynamics, we highlight the non-linear behavior and the relaxation of the emitted voltage amplitude with the input current. Finally the potential applications of spin-torque oscillators are listed.

Part iii presents the first demonstration of brain inspired computing leveraging the non-linear dynamics of the oscillating voltage amplitude emitted by a spin-torque oscillator. In chapter 5 we explain how to perform reservoir computing with a single non-linear node through time-multiplexing. The previous experimental implementations of reservoir computing generally use this method, and classification tasks were successfully solved. However most of these implementations, done with optics, use large components and they are difficult to scale down. In chapter 6, we present the experimental implementation of spintronic reservoir computing used in this thesis, highlighting the physical properties which are used for computation. In chapter 7 we present the classification results obtained on two different tasks which are the sine/square classification and spoken digit recognition, showing in particular a 99.6 % recognition rate for spoken digit recognition.

In part iv, different methods to optimize the spintronic reservoir computing experiment are presented. In chapter 8, we show how to increase the recognition rate by optimizing the experimental parameters such as the rate and amplitude of the input and the applied d.c. current and magnetic field. In chapter 9 we study in the case of the spoken digit recognition, the influence of the time-to-frequency transformation which is performed before using the data in the reservoir experiment. In particular we highlight the critical role that non-linearity plays in such transformations.

In part v we increase the memory of the reservoir using a delayed-feedback oscillator. Chapter 10 evaluates the intrinsic memory of the oscillator through the relaxation and shows that the feedback improves massively the range of the memory. The effect of feedback is closely linked to the operating point. Evidence of a feedback fading memory are obtained by analyzing the response of the delayed-feedback oscillator to a single spike. The memory capacity of this oscillator is also evaluated. These studies are performed varying the operating point of the oscillator. In chapter 11 we study how the feedback memory improves the recognition on sine/square classification on the operating point condition were the classification benefit of the

feedback is highest. We explain this benefit studying how the feedback modifies the emitted oscillating voltage amplitude, which separates the different input cases in the reservoir state. Finally in chapter 12 we show that this benefit depends on the relaxation time of the oscillator and the signal over noise ratio.

Part I

NEUROMORPHIC COMPUTING

1

WHY DO WE NEED NEUROMORPHIC HARDWARE?

1.1 COMPUTATIONAL POWER OF THE BRAIN

1.1.1 *Learning ambiguous situations*

Humans continuously solve pattern recognition problems. For instance, in a fraction of second, one can recognize the same person in different pictures (Figure 1), even though it is nearly impossible to write an explicit criteria for this task. Indeed, each of these pictures is a new situation, similar to the other ones, but not identical. For example, the hair cut, the facial expression, the light, the clothes or the point of view may vary. Somehow, the brain can build a representation of the images, where the same person in different contexts induced the same response. Moreover, this representation is built without supervision, since nobody indicates that it is the same subject in every picture.

In the later example of image recognition, the pattern to identify was spatial (it is for a picture the different colored areas), but the brain can recognize temporal patterns from sensory activity, such as in speech recognition. In that case the temporal position of the phonemes in time is important to understand correctly a word or a sentence. For example, when hearing "mother", the phoneme "mo" should be remembered so when the brain identifies the phoneme "ther", it understands the word "mother" and not for instance "father". The brain can thus process sequences of information (in the later case, the sequence is the temporal signal coming from the ear), due to some memory mechanism.

The example of speech recognition is also interesting because it highlights the necessity of supervised learning. One can understand the signification of a word only if it was told to him. And then, when an other person says the same word, it can identify the same temporal pattern of the word and associate it to the same concept. So after this

Figure 1: Images extracted from the GBU face recognition challenge. Reproduced from P. J. Phillips, A.J. O'Tool 2014 [3].

learning under supervision, this association between a voice signal and a concept can be generalized to new voice signals.
Achieving such pattern recognition with classical algorithms and performance similar to the brain is not possible. For image recognition on the imageNet challenge [4], the best algorithm with explicit rules gave at best 74% [5] recognition rate, when human performance is over 95% [6]. Not only the brain can achieve very complex recognition tasks, but it achieves these tasks rapidly [7] and at very low energy. In average, the human brain works at 25 watts [8]. The brain possesses an architecture specially designed for learning and recognizing patterns at low energy.

1.1.2 *The brain architecture*

If the brain is a subject of interest since prehistory (evidences of ritual trepanations were found on prehistoric skulls), it is relatively recently, in the XIX^{th} century, that it was clearly set as the center of

perception, motion control and cognition. The description of the brain architecture begins in the end of the XIX^{th} century and beginning of the XX^{th} century, notably with the pioneer works of Golgy and Ramon y Cajal, who for the first time visualize the cellular structure of the brain.

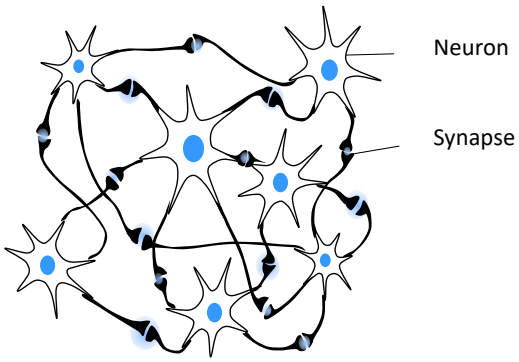

Figure 2: Schematic of of neurons and synapses connections.

The brain is made of long interconnected nervous cells, called neurons, connected through synapses (Figure 2). Synapses are tunable connections and are responsible for the plasticity of the brain, which allow learning. Their mechanism and notably the way they adapt is still not well understood. In this manuscript, we will focus more on the neuron, which can be seen as the "computation" unit of the brain. Neuron cells are able to emit spikes consequently to external stimulus. The architecture of a neuron is shown in Figure 3a. The neuron receives the incoming stimulus which are spike trains in the dendrite (Figure 3b). These stimulus are integrated in the cell body, because they charge the membrane potential of the cell body (Figure 3c). Importantly, this integration occurs with leakage. Once the membrane potential reaches a threshold, the neuron fires a voltage spike in the axon (Figure 3d) called action potential.

The first model describing the integration of a neuron was done in the beginning of the XX^{th} century [9] and it is in the fifties that the first model that describe the mechanism underlying the firing of action potential was proposed by Hodgkin and Huxley [10].

Hudgkin and Huxley describe the action potential of a neuron, modeling the behavior of the permeable membrane of a neuron. This phenomena comes from electro-chemical mechanisms that occur at

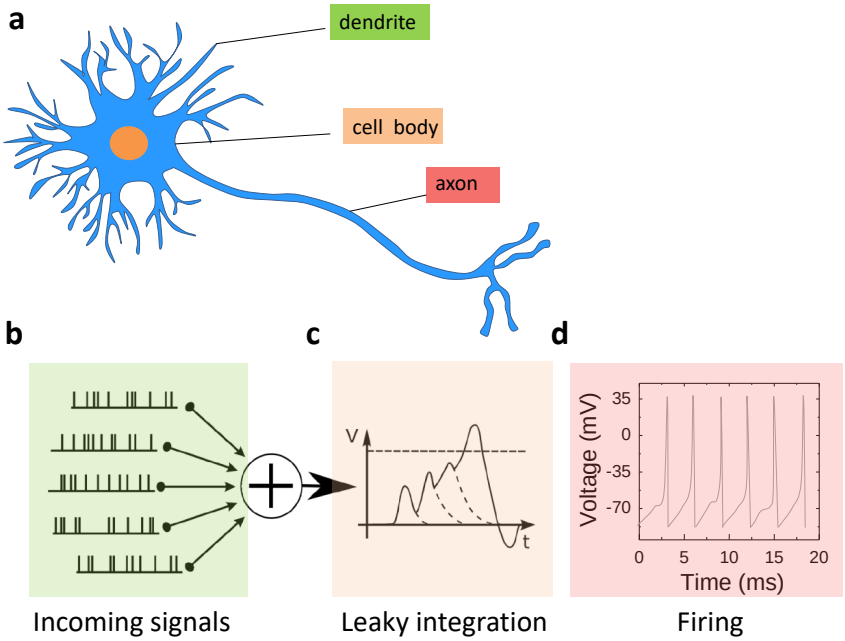

Figure 3: a. Schematic of a neuron. b. Incoming signals from the dendrite c. Leaky integration in the cell body. d. Firing of action potential in the axon.

the membrane of the axon. As described in Figure 4 a, the inside of the neuron is filled in majority with potassium ions, while the outside medium is in majority filled with sodium ions. It thus exists two chemical gradients which should lead to equal the concentrations of ions inside and outside of the neurons. The membrane of the neuron (yellow) is porous, and some ion leak channels (in green for potassium and in red for sodium) allows ionic exchanges. Ion pumps (orange) tend to bring back the potassium ions inside the neuron and expel the sodium ions outside of the neuron.

Because the leakage and the pumping is not the same for sodium and potassium (potassium leaks more than sodium and sodium is more pumped than potassium), there are more positive charges outside the neuron than inside and the membrane has a negative potential (typically -70 mV). When the membrane potential (Figure 5a) increases, due to external stimulus, it can reach a first threshold at

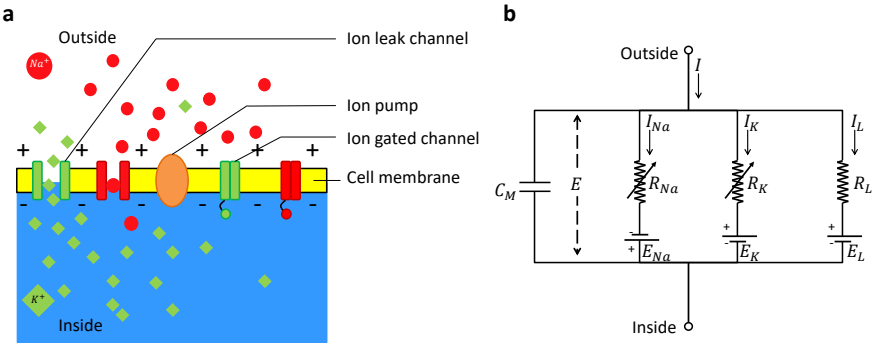

Figure 4: a. Schematic of the axon membrane: the inside of the neuron cell is in majority filled with potassium ions (K^+) while the outside medium is in majority filled with sodium ions (Na^+). The ions migrate through leak and gated channels, while an ion pump maintains the higher potassium concentration inside the neuron. b. Hodgkin-Huxley model: a capacitance models the membrane, the ion leak channels are modeled by a resistance and a generator, each gated channel for potassium and sodium ions is modeled by a gated resistance and a generator. Adapted from [10].

-55 mV. At that moment, the ion gated channels for sodium open (Figure 5b), and sodium ions enter massively inside the neuron (due both to chemical and electrical gradient). It is the depolarization phase. The potential on the membrane increases until it reaches a second threshold around 35 mV. At that moment the sodium gated channels close and the potassium gated channels open (Figure 5c). The potassium ions go massively outside of the cell and the membrane potential decreases. This phase is called repolarization. Finally in a last phase, the gated channels close, and the ion pumps bring the ion concentrations to their initial level (Figure 5d). During this phase called the refractory phase, the neuron cannot spike. In order to describe this spiking phenomena, Hudgkin and Huxley propose an equivalent electrical circuit (Figure 4b). The membrane is modeled by a capacitance C_M, the ion leak channels and ion pumps are modeled by a generator E_L and a resistance R_L. The gated channels are modeled by a generator and a gated resistance E_K, R_K and E_{Na}, R_{Na} respectively for potassium and sodium. The current I represents the external stimulus, which charges the capacitance C_M, until its voltage E reaches the first threshold. The set of equations governing

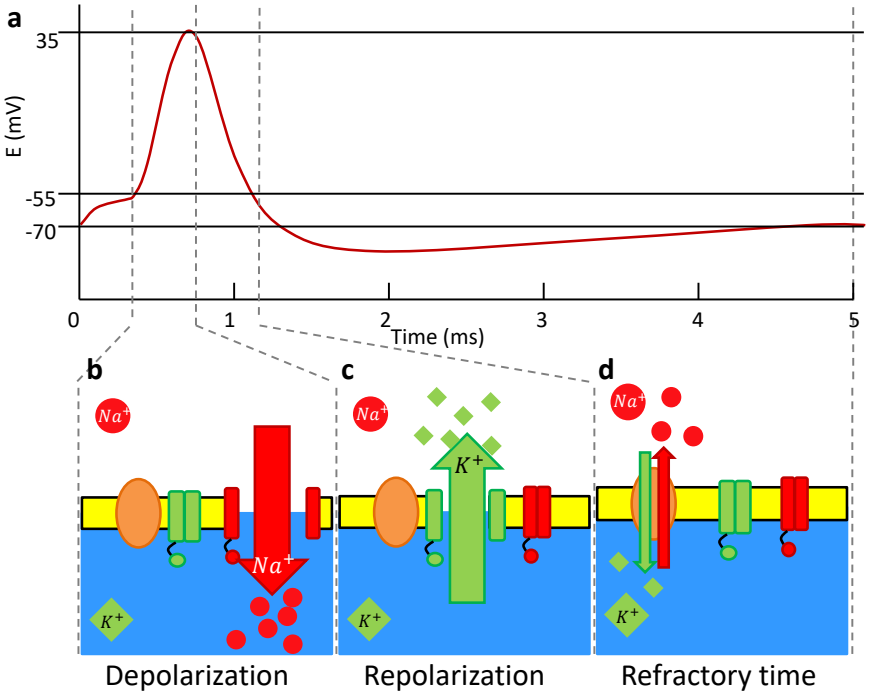

Figure 5: a. Schematic of the membrane potential evolution in time during the firing of an action potential. b. Depolarization phase: the sodium (Na^+) gated channel opens c. Repolarization phase: the sodium gated channel close and the potassium (K^+) gated channels open. d. Refractory time: both gated channels for potassium and sodium are closed. The ion pump brings the membrane potential back to its initial level.

the membrane potential can be found in [10]. R_{Na} and R_K depend on respectively the concentration of sodium n_{Na} and potassium n_K and the variation of these concentrations depends on E.
Importantly in this model, in first approximation, once the membrane potential reaches a threshold (-55 mV), the neuron spikes [11]. So the model of the neuron can be simplified, considering that the neurons integrate with leakage the incoming signal from other neurons in the cell body and once the membrane potential reaches a certain threshold, a spike is emitted in the axon. This model, called leaky-integrate and fire model [9], does not explain explicitly the origin of the spike, nor the refractory period. In comparison with the Hodgkin-Huxley model, it is much easier to compute. The equivalent electrical circuit would be

similar to Figure 4b, but suppressing the two branches with tunable resistances. The membrane potential is described as follows:

$$C_M \frac{E(t)}{dt} = I(t) + I_L(t) \tag{1}$$

where C_M is the capacitance of the neuron membrane, $E(t)$ is the membrane potential, $I(t)$ is the current due to external stimuli and $I_L(t)$ is the leakage current expressed as follows:

$$I_L(t) = -\frac{1}{R_L}(E(t) - E_L). \tag{2}$$

In this case, the membrane potential leaks with a time constant $\tau_M = R_L C_M$. Once the membrane potential reaches a threshold V_{th}, the axon emits a spike, which takes the shape of a Dirac function. The membrane goes back to E_L and stops evolving during the refractory period. The response of the leaky integrate and fire (LIF) neuron to constant inputs is illustrated in Figure 6. The neuron receives two steps

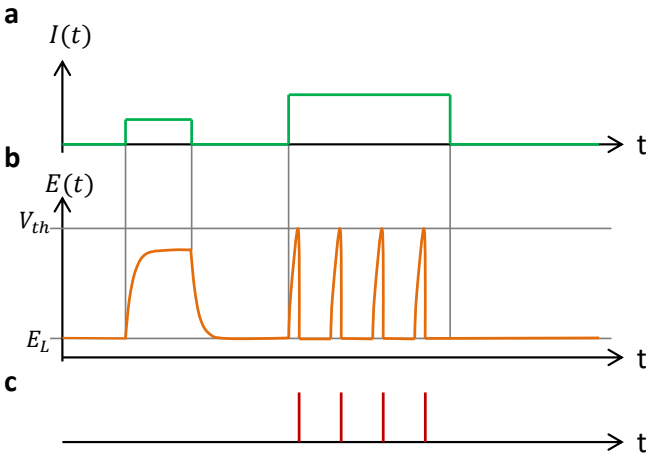

Figure 6: a. External input current $I(t)$ vs time b. Membrane potential $E(t)$ versus time c. Output spike train. Adapted from [12].

of input current with different amplitudes (Figure 6a). The first step is too small for the membrane potential to reach the voltage threshold V_{th} (Figure 6b) and thus during the time of the first current step, no spike is emitted (Figure 6b). After the end of the first input current step, the membrane potential leaks and goes back to its initial level

E_L. The amplitude of the second current step is higher (Figure 6a) and $E(t)$ reaches V_{th} (Figure 6b). Once this threshold is reached, the axon emits a spike (Figure 6c). Immediately after reaching V_{th}, the membrane potential goes back to its initial level E_L and stops evolving during the refractory period, that we will call τ_{ref}. If we consider a constant input I, if $I < (V_{th} - E_L)/R_L$, the neuron does not spike. Otherwise, the neuron spikes periodically, with a frequency f given by:

$$f = \frac{1}{\tau_{ref} + \tau_M log\left[\frac{R_L I}{R_L I + E_L - V_{th}}\right]} \tag{3}$$

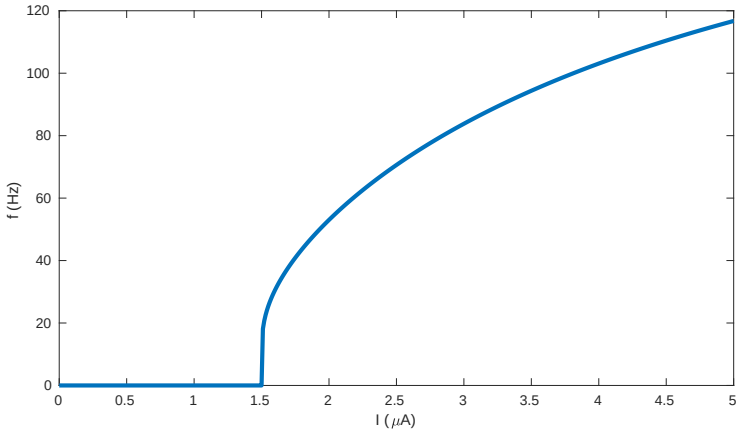

Figure 7: Firing rate of a LIF neuron as a function of the input current. For this plot, $\tau_{ref} = 5$ ms, $\tau_M = 10$ ms, $E_L = -70$ mV and $R_L = 10$ kΩ.

Given this simple LIF neuron model, the neuron can be thought as a non-linear auto-oscillator, which, after the input reaches a threshold $I = (V_{th} - E_L)/R_L$, oscillates at a frequency f, which depends non-linearly of the input value. From a simple point of view, the brain can be seen as an assembly of non-linear oscillators (the neurons), which are interconnected together by tunable connections (the synapses). The key properties of a neuron are its non-linearity, and its memory, which comes from the integration phenomena. In conclusion the brain can achieve complex recognition tasks, because of the memory and non-linearity of neurons and because of the non-volatility and plasticity of

synapses. The first attempts of imitation of the brain were algorithmic. They were motivated by solving tasks such as image and speech recognition that are hard to perform with explicit rule algorithms.

1.2 ARTIFICIAL NEURAL NETWORK ALGORITHMS

1.2.1 *Supervised learning*

The first developments of "neuromorphic computing" were purely algorithmic. The main examples of these "neuromorphic algorithms" are artificial neural networks. The goal of these algorithms was to take inspiration from the brain to solve problems where it is not possible to write explicitly a discrimination rule. They are part of a larger algorithm field denominated as "machine learning". These methods allows to learn, from the structure of the data, an implicit rule for solving a problem. Machine learning algorithms possess inner parameters, and by tuning them appropriately, depending of the data structure, can find an implicit rule for solving the problem.

Two main classes of algorithms can be distinguished: unsupervised and supervised learning. The first one refers to the algorithms which solve problems by detecting automatically regularities in the structure of the input data, without having other knowledge a priori on the data. An example of unsupervised learning is for instance clustering, where the algorithm detects inputs that are similar.

In this thesis, we will use supervised learning, which encounters recently an important success with the development of deep learning [13]. In this case, the algorithm is divided in two distinct phases: learning and inference. During the learning phase, examples are presented to the algorithm, associated with the correct output. For example, for image recognition, algorithms are fed with images and the correct label of the image (i.e. what represents the image). The knowledge of the correct output on these examples is specific to supervised learning. At the end of the learning phase, the tunable inner parameters are fixed, and during the inference new examples, without the label, are presented to the algorithm. During inference, the algorithm applies what was learned during training.

Supervised learning originates from the fifties with the perceptron algorithm which allows to learn the parameters of linear combinations

[1]. After the development of computational power, it encountered new successes after the eighties. Supervised learning includes notably support vector machine (SVM), kernel methods, and artificial neural networks. This last class will be used in this thesis. The algorithm re-

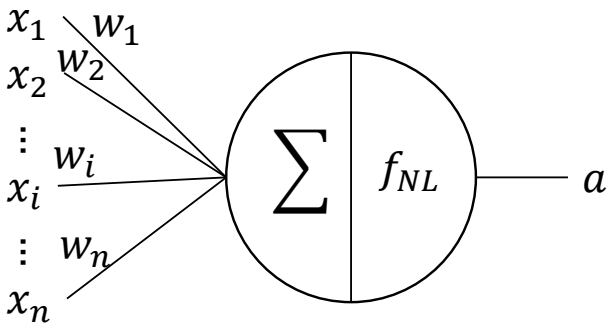

Figure 8: Schematic of a formal neuron: the different inputs x_i are multiplied by a connection weight w_i in the synapse. The formal neuron sums the inputs and applies a non-linear function to the result of this sum.

produces the response of a network of non-linear units, called neurons, interconnected together with tunable connections called synapses. Importantly these neurons are much more simple than biological neurons. The formal neuron achieves the summation of the inputs and applies to this sum a non-linear function called activation function. This activation function is in a way analog to the firing rate of the neuron seen in the previous section. Classical activation functions used are for instance arctan, sigmoid or rectified linear units (ReLu) (Figure 9).

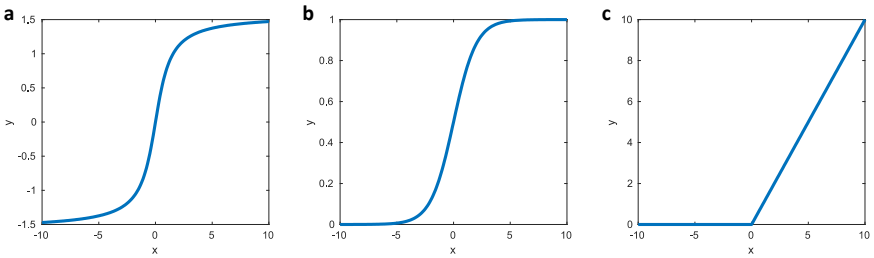

Figure 9: Classical activation functions: a. Arctangent b. Sigmoid c. Rectified linear unit (ReLu).

In the case of artificial neural networks, the algorithm learns to solve a problem by adjusting the connection weights during the learning

phase. During the inference phase, the weights are fixed, and the input encounters a series of non-linear transformations, which has been learned. Artificial neural networks can be classified in two categories that are the feed-forward and recurrent networks.

1.2.2 *Feedforward neural network*

Feed forward neural networks refer to networks made of several layers of neurons which are connected one after the other. This architecture was popularized in the 2000's and achieve state of the art performance in web advertizing or image recognition. The interest of feedforward networks is that it is possible, even with a large number of neurons and connections, to train a model. The algorithm widely used for this training is backpropagation, introduced in the eighties [14]. The main examples of feedforward neural networks are fully connected networks, and convolutionnal networks. In fully connected network, each neuron of a layer is connected to all the neurons of the next layer (Figure 10). These networks are complex to train when the layers have

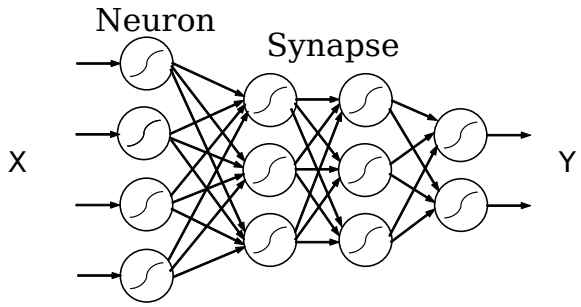

Figure 10: Schematic of a fully connected network. The circles represent the formal neurons which compute a summation and a non-linear activation. The arrows represents the synaptic weights that are tuned for computation.

numerous neurons. Indeed the number of connections (ie parameters to train) evolves typically like the square of the number of neurons per layer. In practice, when the data to process has already a large dimension, like for example images, the fully connected approach requires too many parameters to train. For image recognition, the actual state of the art networks are convolutional neural networks [15]. The trans-

formation from one layer to another one is done by convolution with a filter forming a feature map (Figure 11). Importantly, the filter has

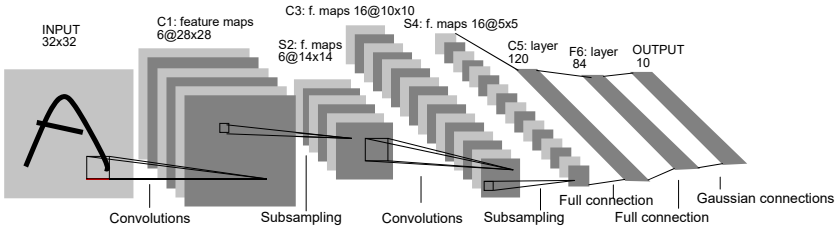

Figure 11: Architecture of the convolutional network "LeNet-5". Each plane is a feature map, i.e. the results of a convolution with a filter. Extracted from Lecun et al [15].

a small dimension and only the parameters of the filter are tuned for learning. This approach reduces drastically the number of parameters to tune, when compared with the fully connected approach, and is well suited for translation-invariant problems such as image recognition. Training is also performed with back-propagation algorithm. State of the art performance were achieved with such approach on image classification. For instance the inception network achieved a 97% recognition rate on imageNet challenge [16], which is better than human performance for this task. Nevertheless, this algorithm can be quite computational costly. In the case of the inception network, the network counts 35 million parameters that need to be trained and one single inference requires 19 billions operations [17]. If feed-forward networks give impressive results for image recognition, they are not well suited to process sequences, such as speech recognition, because they do not account for the order. Dealing with such problems requires recurrence in the network.

1.2.3 *Recurrent neural networks*

When it comes to process sequence data like words in a sentence, or the trajectory of an object, the most suited neural network architectures are recurrent neural networks.

In recurrent neural networks, the connections between neurons form inner loops in the network, where the information cycle and thus is

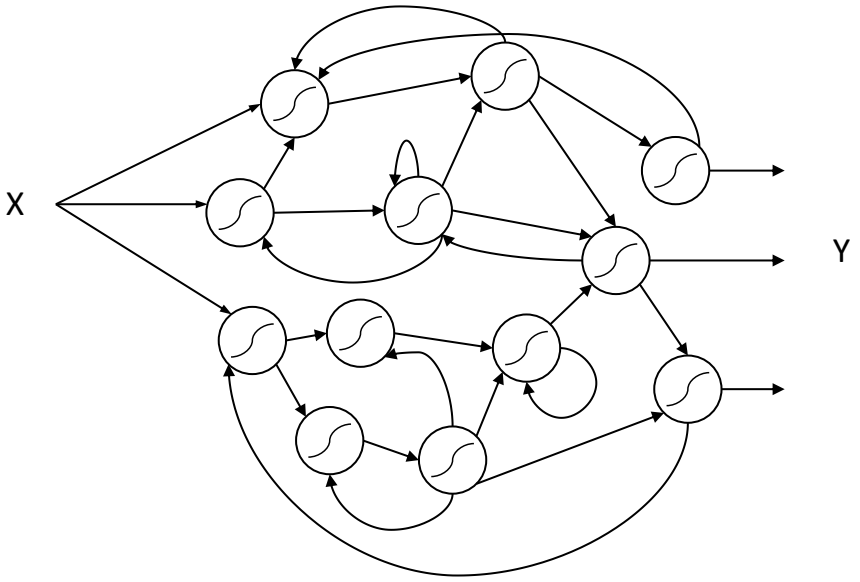

Figure 12: Schematic of a general recurrent neural network.

selectively stored (Figure 12). Such networks hold in their architecture a memory and are well suited to process sequence information where the order of the elements is important (such as sentences in speech recognition for example).

Two particular architectures, long short term memory (LSTM) [18] and bidirectional recurrent neural network (BRRN) [19], achieved state of the art results to process sequence data such as speech recognition [20, 21]. As an example, human performance for word transcription [22] were achieved, using a mixture of convolutional and LSTM networks.

But in the general case, it is extremely difficult to train recurrent neural network, and the training algorithm may not converge even after an infinite time of computation [23, 24]. Even for LSTM and BRRN the training algorithm used, which is back-propagation through time [25], is generally very greedy in term of computational power and time. Reservoir computing [26, 27] is a recurrent network much easier to use, because it does not require to train the recurrent connections. This particular recurrent network, that we will use experimentally in the manuscript, will be described in much more details in chapter 5.

1.3 LIMIT OF CLASSICAL COMPUTERS FOR MACHINE LEARNING

Artificial neural networks are extremely powerful tools, and, running such algorithms, human performance were reached and even overcome on image recognition and speech recognition tasks. Nowadays these algorithms are massively used for big data analytics such as online advertisement, finance and banking, health care, autonomous driving, technology. Such technologies are believed to spread soon to other economical sectors, and notably in the growing sector of the internet of things [28]. They are brain-inspired because they take inspiration from the general brain architecture, keeping the plasticity of synapses and the non-linearity of neurons. But these algorithms require many computation operations (typically few tens of giga-operations for only the inference on [17]) and they are run on classical architecture computers. The Von-Neuman architecture is at the basis of the computers we use, where the computational unit (CPU) is explicitly separated from the memory part [29] (Figure 13).

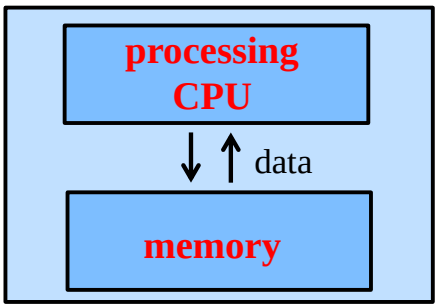

Figure 13: Schematic of the Von Neuman architecture.

To perform a task, a computer follows a sequence of instructions, which involves generally to retrieve data in the memory, perform a computation and store the result in the memory. This architecture is sequential, data is processed and stored one by one. This sharing of the bus between memory and processing unit is called "Von Neumann bottleneck" [30]. So at each computation step, the data makes a round trip between memory and processor. This bottleneck limits the speed of the computation, because the time to retrieve an information in the memory is much longer than the time of an operation. Importantly, the

cost of storing and retrieving data in the memory consumes much more energy than the operation itself [31], so the "Von Neumann bottleneck" is also an issue in term of energy. Neural networks have a massively parallel architecture, and they are modeled with many parameters (typically tens to hundreds of millions [17]) which need to be retrieved. It represents huge flows of data which need to be moved back and forth from memory to processing. For neural network algorithms, the Von Neumann architecture is especially not well suited. First improvements were done using parallel computation, with multi-cores (so multiple Von-Neuman architectures) and graphical processing units (GPU), which speed up the computation enabling for instance complex image recognition [32]. The field programmable gate arrays (FPGA) allow in some cases better energy consumption with comparable results on some tasks [33]. Finally companies are now developing chips specialized for machine learning operations, such as the TPU from Google, the A11 bionic neural engine from Apple, the Holographic Processing Unit from Microsoft or the NPU from Nvidia.

Despite orders of magnitude gains in term of energy consumption when compared with a single CPU unit, the energy consumption is still orders of magnitude higher than the one of the brain. Indeed the brain performs well because the processing due to neurons is entangled with the memory hold by synapses. In the end, the information does not need to travel long distances. A promising research perspective is thus to take inspiration of the brain, not only for the algorithms to compute but also for the physical hardware which performs the computation.

2

STATE OF THE ART OF HARDWARE NEURONS FOR NEUROMORPHIC COMPUTING

As it has been seen in the first chapter, artificial neural network algorithms are very powerful, and are becoming more and more essential in many industrial fields. However, since these algorithms require computing the response of millions of neurons, running them on classical computers is energy costly, even with multi-core CPUs and GPUs approach. Dedicated hardwares, which emulate synapses and neurons are thus required. To build chips with a reasonable size would typically require a neuron density superior to $10^6 - 10^8\ cm^{-2}$. In practice such requirement for a fully analog chip would require components which are smaller than 1 μm.

In this chapter I review the main attempts to build scalable neurons for brain-inspired chips. As neurons can be considered as non-linear oscillators (see chapter 1), thus oscillators based on nano-components are also presented. A special focus is done on spiking neurons. We do not review in details the attempts for building hardware synapses. It is a very large field, which is outside of the perimeter of this manuscript.

2.1 COMPLEMENTARY METAL OXIDE SEMICONDUCTOR HARDWARE

The first attempts to build brain-inspired hardware architectures were conducted in the complementary metal oxide semiconductor (CMOS) community.

The term "neuromorphic" itself was introduced by Mead et al in the field of very large scale integration (VLSI) in the late eighties [34, 35]. In neuromorphic computing, there are two main motivations for building hardware artificial neurons: for a fundamental understanding of the human brain [36, 37] or for computing at low energy [38, 39]. In our case it is more the last approach which is our interest. Also, we will limit this state of the art to oscillators and spiking neurons. For

emulating spiking neurons, different technical solutions have been chosen: analog implementation [40, 41], mixed analog numerical implementation [37], processor driven [36], digital approach [38, 39]. We also present in a first subsection the CMOS oscillator, which could emulate neurons as non-linear oscillator, even if they were not always initially designed for this purpose.

2.1.1 *CMOS ring oscillators*

CMOS ring oscillators are not initially designed to be used for neuromorphic computing, but since they are non-linear oscillators, they could be potentially used for this purpose. Ring oscillators are made of an odd number of "not" logic gates (Figure 14). Because

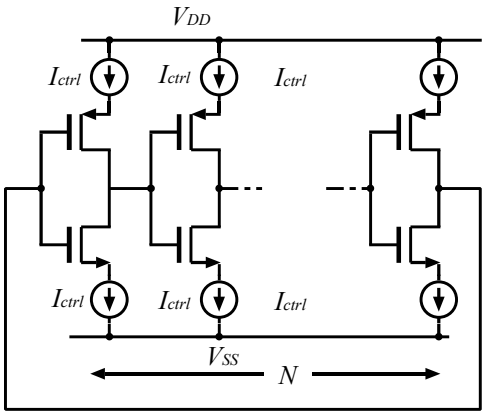

Figure 14: Schematic of a CMOS ring oscillator. Extracted from Yan et al [42].

of parasitic capacitive effects [42], the not gates switch with a delay time. The odd number of not gates combined with the switching delay induces self sustained voltage oscillations with a frequency $f = 1/N\tau$, where N is the number of not gates and τ is the switching delay. Interestingly, τ can be tuned changing I_{ctrl} the current crossing the not gates (Figure 14) [42]. With additional circuits, the range of frequency of such oscillator can be increased [43, 42, 44, 45] and high frequencies can be reached [46]. This class of oscillators can operate in a large range of frequency: 40Hz-380 MHz [42], 660MHz-1.27GHz [43], 100MHz-3.5GHz [47], 1.5 GHz [48], 8-16 GHz [49]. In particular, a single

oscillator can have a wide range of tunability [42, 43, 49]. The power consumption and the energy spent per oscillation also depends on the range of frequency. Finally, the footprint of these oscillators depends on the design but is typically 10 μm^2 [48] to 10^5 μm^2 [49]. For the moment, even if synchronization of ring oscillators was demonstrated [49], no demonstration of computation was performed with such oscillators.

2.1.2 *Spiking neurons*

The first solution for analog neurons consists in building dedicated circuits for all the neuron functionalities which are the leaky integration due to the conductance dynamics, the spiking, the reset refractory period and the frequency adaptation [41, 50]. In particular, the conductance dynamics of a neuron can be described by a first order equation and therefore, it can be emulated by first order filter CMOS circuits [40, 41].

The first CMOS analog neuron was built in the early nineties by Mahowald and Douglas [40]. This circuit reproduces the behavior of a Hodgkin-Huxley neuron. Other conductance based silicon neurons were proposed later on [51, 52, 53, 54]. But these circuits are complex and thus require a significant number of transistors and bias voltages or currents [50].

Therefore, simpler circuits emulating LIF neuron model were developed. The most simple approach is to use a capacitor for the integration of the neuron and when the capacitance voltage reaches a certain threshold, it activates a simple spiking circuit [55]. A classical circuit for the spiking threshold is for instance the Axon-Hillock circuit [34]. However this implementation dissipates a non negligible amount of power [55]. More power efficient LIF neuron circuits have been developed in [56, 57, 58, 55, 59].

Finally, intermediate complexity models of neurons (such as Izhikevich model [60], adaptive integrate and fire model [61], Morris-Lecar model [62]) have been emulated in CMOS for instance in [63, 64, 65, 66, 67].

Such analog CMOS neurons can reach satisfying energy consumption (50 $pJ/spike$ [63], 8.5 $pJ/spike$ [59]) and power consumption (1.5 nW in [59]). Most of these silicon analog neurons are developed for real time modeling of biological neurons so their frequency is generally

of the order of tens to hundreds of Hz (frequency of a biological neuron). In [59], the neuron was accelerated to 1 MHz. The main draw-back of silicon neurons is the surface they use with at best 7 μm of lateral size [63].

2.1.3 *Neuromorphic chips*

Several chips were built in order to emulate the behavior of large neural networks in energy efficient fashion. The SpiNNaker chip [36], the HICANN chip [68, 69] and Neurogrid [37] were designed for more fundamental applications. The goal is to emulate the behavior of biological neurons in order to understand the brain activity. In particular in [36], two different models of neurons can be implemented [9, 60]. These approaches can be seen as "energy efficient" when compared with brute force simulation of neurons on super computers, but they remain much more energy intensive than a real biological brain. The spiNNaker computer is based on custom micro-processor approach, using ARM processors. Figure 15 shows the platform for 500 000 cores, showing that even though it performs better in terms of energy than super computers, it is far away from the dimensions or the energy efficiency of the brain.

Figure 15: The 500 000 cores SpinNNaker Human Brain platform. Extracted from [70].

On the other hand the Neurogrid and the HICANN chips use a mixed digital-analog implementation, with analog neurons. Typically the size of the neuron is 50×50 μm^2 [37] for Neurogrid, and it emulates simple

sub-threshold LIF neurons. For HICANN chip the analog neuron is $150 \times 10\ \mu m^2$ and it emulates an accelerated adaptive exponential neuron. Industrial actors also developed neuromorphic chips, but more for artificial intelligence tasks such as object detection. The "True North" chip [38] was unveiled by IBM in 2015. This chip is fully digital and uses time-multiplexed neurons. Each neuron block is typically 10 μm per 100 μm large (Figure 16) and emulates in time the response of 256 neurons. Thus with time multiplexing and approximatively 4000 neuron blocks, the chip emulates the behavior of one million neurons. The chip can perform successfully image detection, with 176 000 less energy than a regular processor approach (but it is the energy consumed by a processor running the exact same spiking neural network, it does not correspond to the energy cost of running object detection with the standard convolutional networks seen in the chapter 1), and 769 less energy than with a SpiNNaker approach [38]. However

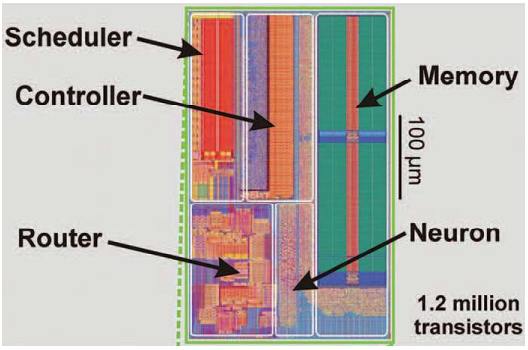

Figure 16: The physical layout of True North core in 28-nm CMOS fits in a 240 μm-by-390 μm footprint. A memory (static random-access memory) stores all the data for each neuron, a time-multiplexed neuron circuit updates neuron membrane potentials, a scheduler buffers incoming spike events to implement axonal delays, a router relays spike events, and an event-driven controller orchestrates the core's operation. Extracted from [38].

a main draw back of this chip is that it can only perform inference and not learning. In 2018, Intel unveiled its fully digital chip "Loihi" [39], which, when compared with the True North chip, can perform learning. The chip possesses 128 neuromorphic cores on a 60 mm^2 chip, with each core emulating in time 1024 neurons. So in total the chip only emulates 130 000 neurons. This chip could solve LASSO optimization

problem, winning three orders of magnitude in energy-delay product, when compared with conventional approaches. The size of the neuron block is not mentionned in [39], but the neuron density of the Loihi chip is worse than the one of True North even if each neuromorphic core emulates in time more neurons than the True North chip (1024 neurons for Loihli and 256 neuron for True North). Both of these digital neuromorphic chips emulate LIF neurons. In conclusion, with CMOS technology, analog non-linear oscillators and analog neurons can be built. Large chips using mixed-analog-digital and fully digital technologies were developed. But CMOS technology encounters a scalability problem, with components with a lateral size which is few μm (for the most simple CMOS ring oscillators [48]) to few hundreds of μm (for the digital neuromorphic cores [38]). This observation motivates trying to replace parts of CMOS circuits by nano-devices, for which the physics implements important functionalities of the neuron.

2.2 RESISTIVE SWITCHING DEVICES

As seen in the previous chapter 1, neurons can be modeled by electronic circuits with switching resistance (Hodgkin-Huxley model) or by a more simple circuit which integrates the signal and resets each time a threshold is reached (LIF model). These equivalent circuits can be built in hardware using resistive switching devices. Resistive switching devices can be nanometric components and thus a single component can replace a whole CMOS circuit to implement some functionalities of a neuron. This resistance switching devices belong to the class of memristors [71].

2.2.1 *Different types of resistive switching*

The mechanisms underlying the resistance switch can be classified in 4 categories: phase change, red/ox reaction, charge effects and purely electronic effects.

Even though it is not the main subject of this thesis, it is important to mention that memristors are extensively studied to implement hardware synapses in neural network. Indeed, naturally memristors,

which are analog valves for currents, behave like synapses. Many different kind of memristive synapses have been studied so far including ferroelectric [72, 73], spintronic [74, 75], red/ox [76, 77] and phase change [78, 79].

More recently, analog neurons were built using certain kinds of memristors. These resistive switching devices used purely electronic effects (Mott transition [80, 81, 82, 83, 84]) and red/ox reactions [85, 86, 87].

For neuron applications, memristors must have different properties than for building synapses. Indeed it should exhibit a negative resistance differential and a resistance switch behavior [88]. So when the voltage at the electrodes of the memristor increases, the resistance of the device should decrease (after a threshold value) and when the bias voltage is removed, the low resistance state should disappear. This last property is different from the requirement for a synapse which should keep the resistance information even when the bias voltage is removed.

In the case of [80], the resistance switch comes from a voltage-favored transition from Mott insulator to metal phase in NbO_2 materials [89, 88]. Interestingly, the metal phase is metastable and disappears when the bias voltage is removed. Figure 17 shows a schematic of the energy diagram for the stable Mott insulator (MI) state and the meta-stable conductive metal (CM) state.

In [81, 82, 83], VO_2 material is used. The reason of the resistance switch is still not exactly well understood: it could be either a Mott to metal phase transition or an electron-phonon interaction [81].

In [85, 86, 87], for red/ox memristor, the resistance switch is due to atomic migration. In particular for [85] in TiO_x and TaO_x materials are used and for [86] in WO_x, the resistance switch is explained by migrations of oxygen vacancies which induce the formation of a conductive filament. In [87], SiO_xN_yAg material is used and the resistance switch comes from the migration of Ag atoms. Figure 18 illustrates the migration of oxygen vacancies that occurs in some types of red/ox memristors.

Importantly, the choice of the resistive switch material changes the range of resistance, the switching time, the energy required to switch and the endurance of the device. Typically, a red/ox switch can achieve 10^6 cycles with 10 fJ per switch and takes 50 μs to go back

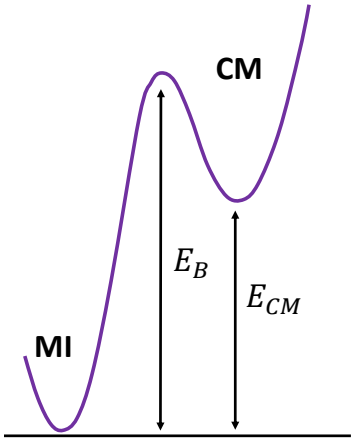

Figure 17: Schematic of the energy diagram of the conductive metal (CM) and Mott insulator (MI) states and the energy barrier. Adapted from [90].

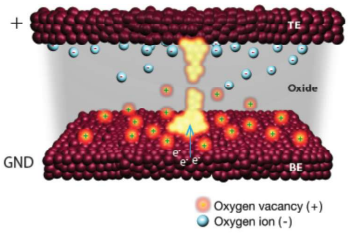

Figure 18: Schematic of a silicon oxide memristor. The change in resistance is due to the motion of oxygen vacancies. Extracted from [91].

to its stable state [87] and a Mott insulator can achieve 10^9 cycles with 100 fJ per switch and 2.3 ns to go back in the stable state [88].

2.2.2 *Memristors for spiking neurons*

The most natural idea to build a hardware neuron with resistive switch devices is the Hodgkin-Huxley model which explicitly represents the resistance switch in the equivalent circuit (see Figure 4b). Such devices have been built in hardware by Pickett et al [80]. In such approach, a single device can replace a whole circuit of a conductance based silicon neuron (i.e. first order filter mentioned in 2.1.2). It is not exactly the Hodgkin Huxley model equivalent circuit, because

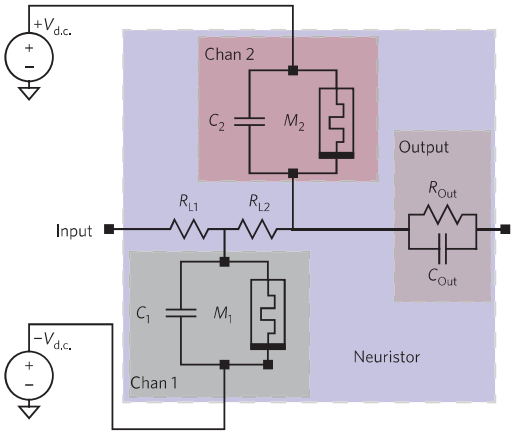

Figure 19: Circuit diagram of the lumped neuristor. The channels consist of Mott memristors (M_1 and M_2), each with a characteristic parallel capacitance (C_1 and C_2, respectively) and are biased with opposite polarity d.c. voltage sources. Extracted from Pickett et al [80].

the two memristors M_1 and M_2 are identical, on the contrary of the variable impedances R_{Na} and R_K in Figure 4. This circuit is called a "neuristor", as a reference to the work of Crane et al [92], and captures the main features of the action potential: threshold-driven spiking, lossless spike propagation at a constant velocity with uniform spike shape and a refractory period. The demonstration was done with Mott memristors which switch rapidly with low energy consumption [88]. The spiking frequency depends on the input current value but also on the capacitance values C_1 and C_2. The spiking frequency of such circuit is typically of the order of few tens of kilohertz. Very recently, a similar approach was chosen in [84], with VO_2 memristor to emulate an Hodgkin-Huxley neuron.
Other research teams achieved more simple circuits closer to a LIF neuron [82, 83, 81, 85]. For such circuits, the building block allowing the sustained oscillations when a d.c. current is applied is represented in Figure 20a.

Initially, the resistive switching device M is in a high resistance state R_{off} (off state) (Figure 20c). When a constant current is applied, the capacitance C charges with a characteristic time $R_{off}C$ until M reaches its voltage threshold (Figure 20b). Then the resistance of M

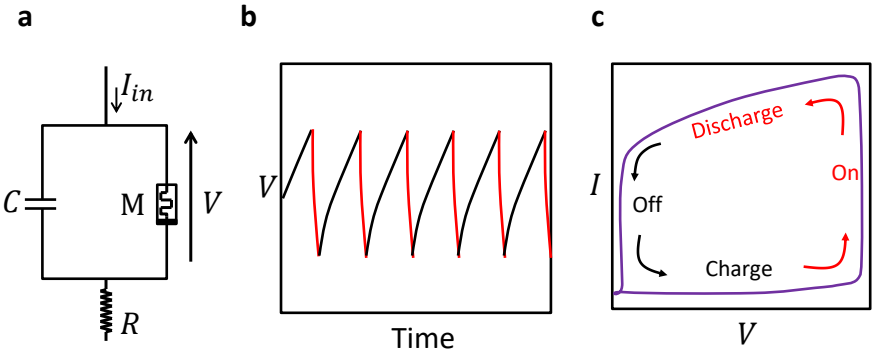

Figure 20: a. Memristive auto-oscillator: a capacitance C is in parallel with resistive switching device M. A load resistance R is connected in series. b. Schematic of the voltage vs time c. Schematic of the current voltage curve for M.

drops to a lower state R_{on} (on state), and for adequate values of R, the capacitance discharges more rapidly (with a characteristic time $R_{on}C$) and the resistance switching device M goes back to it initial higher resistance state (off state). The behavior of such oscillator is similar to the membrane potential of LIF neuron (see Figure 6b) but without refractory period. The emission frequency of such devices reported in [81, 85] are typically of the order of tens to hundreds of kilohertz. Sharma et al [85] proposed also a similar device but ballasted with a transistor. In that case higher frequency can be reached (up to hundreds of megahertz).

Several attempts to solve classification tasks with memristive neurons were performed recently [86, 87]. All these approaches did not use auto-oscillators as neurons. The approach of Du et al [86] uses the non-linear dependence of the memristor resistance with voltage, as an equivalent to the activation function of the software artificial neural networks (Figure 9). They used a mixed spatial and time-multiplexed approach with 88 memristors and tested their device on the classical MNIST image recognition task. They obtained 88.1% success rate with 88 "non-linear nodes", which constitutes a moderate success rate on this database. Indeed a simple logistic regression gives 90 % success rate and a convolutional network gives 99.8 % error rate [93]. Finally Wang et al [87] use memristors in parallel with a capacitor to have integration and relaxation. These analog neurons are submitted to

series of square-shaped voltage inputs. The neuron output considered is the current flowing through the memristor. They could implement ReLu activation functions with such devices (Figure 9). They built a 8 input channels-3 output channels chip, using memristors to emulate neurons and synapses and recognized simple patterns. This chip emulates a simple convolutional layer. They tested learning on simple patterns "U", "M","A" and "S" encoded on 4×4 pixels. This task is quite simple and the success rate is not indicated.

Several examples of analog neurons have been built with resistive switching devices. Some of them are non-linear oscillators [88, 81, 85, 84], some others just emulate activation functions [86] or perform integration and fire behavior if they receive spiking inputs [87]. There were some recent attempts to perform classification tasks with memristive neurons [86, 87], but the performance is still far from the state of the art. Moreover a major drawback for the use of resistance switching devices for computing is their limited number of cycles, which is in average 10^9 for Mott-metal transition switch and 10^6 for the red/ox switch.

2.3 JOSEPHSON JUNCTIONS

Josephson junctions are made of two superconductors separated by a thin non-superconductive material barrier [94]. The electrons of each superconductor are described by wave functions with specific phases. Interestingly, if the barrier is thin enough, the wave functions overlap, and a super-current which depends on the phase difference appears in the barrier. If a bias voltage is applied to the junction, the super-current oscillates with a frequency which depends on the bias voltage value. Thus Josephson junctions have been extensively studied as non-linear tunable oscillators [95, 96, 97, 98, 99, 100, 101, 102]. Such oscillators can reach very large frequency ranges. Indeed a frequency range from 100 kHz to 1 THz was reported in [98]. In order to improve the power of the emission and reduce the bandwidth, oscillators are generally built with coupled Josephson junctions [96, 99, 101, 100].

More recently, an analog artificial neuron designed using Josephson junctions was first simulated in [103] and then built experimentally in [104].

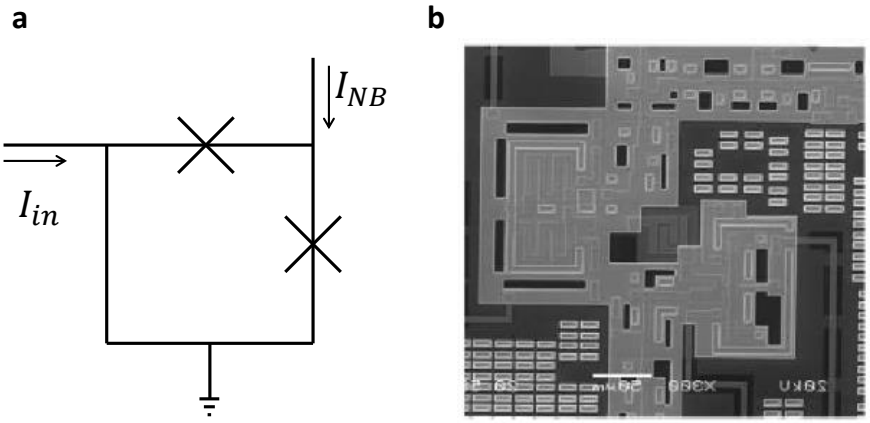

Figure 21: a. Circuit diagram of the Josephson neuron. Two Josephson junctions are connected in a loop. The neuron is biased by two currents I_{in} and I_{NB}. b. Optical microscope picture of the neuron and synapse combination circuit. Extracted from [104].

The Josephson neuron is made of two Josephson junctions connected in a superconducting loop (Figure 21a). Even though the size of the neuron demonstrated in [104] is tens of μm large (Figure 21b), the circuit may probably be scaled down, since much higher densities of Josephson junctions have been demonstrated with a million of junctions per squared centimeter [105]. Moreover these neurons spike rapidly (typically 25-50 GHz) with very low energy (10^{-17} J). In [104], the coupling and synchronization of two neurons was demonstrated. The main drawback of Josephson junctions is that they do not work at room temperature. For high critical temperature (T_c) superconductor, $T_c = 90$ K and a nitrogen cooling system is still required. The Josephson neuron demonstrated in [104] works at $T = 4$ K, with a helium based cooling system. Moreover, the efficiency of such neuron to solve cognitive tasks remains yet to be demonstrated experimentally.

2.4 ELECTROMECHANICAL OSCILLATORS

Finally, electro-mechanical oscillators are another class of potential non-linear oscillators that could emulate hardware neurons. These devices incorporate a mechanical cantilever that can be scaled down to 20 nm height and width [106]. The vibration of the cantilever is

read using piezo-electric materials. In order to obtain self-sustained oscillations, a delayed-feedback loop can be added to the system [107]. Resonant frequencies over 1 GHz were reported in [108], and tunable resonant frequency in a typical range of 50-150 MHz is reported in [109]. Such oscillators can couple and synchronize [110, 111]. These oscillators have been used notably for sensing [112, 113] and high-frequency processing [114, 113]. More recently, [115] proposed to use coupled electro-mechanical oscillators for computation and in [116], an experimental demonstration of reservoir computing with a single electro-mechanical oscillator was performed, solving benchmark tasks such as spoken digit recognition. Also for this demonstration, the mechanical oscillator used was micrometer-scale.

2.5 CONCLUSION

CMOS is a mature technology, which explains that the most advanced neuromorphic devices in terms of task solving were developed with this technology. In particular the True North chip could solve complex tasks such as object detection, with much less energy than standard approaches. But emulating the functionalities of a neuron requires complex or energy intensive transistor circuits which limits the analog approach. Indeed the smallest CMOS neurons have still a 7 μm lateral size. To meet the neuron density requirements, the True North chip uses neuromorphic cores which emulate in time 254 neurons each.

There is thus a need for alternative nano-components which could emulate the functionalities of a neuron. The first approach presented are the resistive switch memristors. Even though these components can be scaled down to nanometric dimensions, they are generally used in a circuit with additional larger components. For instance they are generally used in parallel of capacitance to have an auto-oscillator behavior. Moreover the main draw-back until now is the limited lifetime of such devices, which for now hardly exceed 10^9 cycles in average, which would limit the lifetime of such neurons to few hours. If the energy efficiency of these devices to emit a spike is satisfying, the power dissipation may be too high for dense integration. Moreover, the attempt to solve cognitive tasks are still preliminary.

Josephson junction based neurons have also been proposed. If such device downscales under a micrometer, the actual demonstration is still 50 μm large. Compared with other approaches they work fast (100 GHz), with low energy (10^{-14} $J/spike$), but require cooling. Moreover the latest demonstration only showed synchronization and no cognitive tasks were solved yet.

Finally electro-mechanical oscillators can be scaled-down under the micrometer. Such oscillators can interact with each other and they work at low energy. It is only very recently (2017) that they were investigated for brain-inspired computing. The first attempt to solve a task experimentally is made in 2018, but these results are obtained with a device which is larger than a micrometer.

In this manuscript we propose to use spin-torque oscillators as potential hardware neurons. In order to meet the component density required for developing a brain-inspired chip, such device should be smaller than a micrometer and emulate by itself the important functionalities of a neuron. It means it should behave as an auto-oscillator, which has memory and non-linearity. Moreover they should be able to couple with each other, they should work at low energy and they should have a long lifetime. Finally the viability of such device should be demonstrated by solving a cognitive task.

Part II

SPIN-TORQUE OSCILLATORS FOR BUILDING HARDWARE NEURONS

3

PHYSICAL PHENOMENA INVOLVED IN SPIN-TORQUE OSCILLATORS

In this chapter, we present the physical phenomena underlying the working principle of a spin-torque oscillator, which are namely magneto-resistive effect and spin-torque effect.

3.1 ELECTRON TRANSPORT IN FERROMAGNETS

Spin-torque oscillators (STO) belong to the class of spintronic devices. Spintronics leverages the electron spin information instead of the charge information in classical electronic. Therefore, transition metals are extensively used in this area because in these materials, the transport of electrons depends on their spin.

Electronic transport properties of ferromagnetic materials were described by Mott in 1936 [117]. Ferromagnetic materials have a spontaneous magnetization because a majority of the electron spins in this material are pointing in the same direction. The alignment of the spins comes from the atomic-like exchange interaction between spins in the material [118]. Rare earth 4f materials and 3d transition metals such as iron, cobalt or nickel and their alloys exhibit ferromagnetic properties at room temperature. In rare earth materials, the hybridization between atoms is weak so it does not overcome the exchange interaction. But the conduction electrons are localized and the resistance is high, which limits their application for spintronics. So these materials are more used as permanent magnets [119]. The 3d transition metals have both strong exchange interaction and strong inter-atomic coupling. They also exhibit ferromagnetism and have lower resistance. Interestingly, ferromagnetism comes from the localized 3d electrons and influences the transport of 4s conduction electrons.

The electronic band structure of transition metals is schemed in Figure 22a. In this simplified model the 4s electrons are responsible for the conduction while the 3d electrons are localized and responsible for

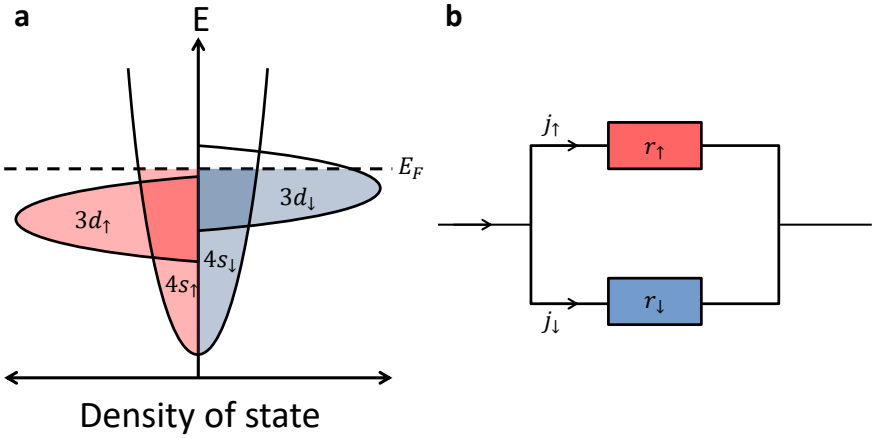

Figure 22: a. Schematic of the spin dependent band structure of 4s and 3d electrons of transition metal such as cobalt. b. Equivalent simplified two-channels model. Adapted from [120].

magnetic properties of the material. The exchange interaction between electrons induces a splitting of the 3d electron band, depending of the spin. By convention the majority spins will be designed ↑ and the minority ↓. The exchange interaction is favorable for $3d_\uparrow$ electrons and thus their band is lower in energy when compared with $3d_\downarrow$ electrons. The $4s$ conduction electrons are not affected by the exchange interaction and thus their band structure is symmetrical. Because of the spin dependent splitting of the $3d$ band, the overlap between $3d$ and $4s$ at the Fermi level is more important for the electrons ↓ than ↑.

It results that the probability for $4s_\downarrow$ to be scattered by $3d_\downarrow$ is higher than the probability for $4s_\uparrow$ to be scattered by $3d_\uparrow$, and thus the $4s_\uparrow$ are better diffused through the ferromagnet than the $4s_\downarrow$. This spin dependent current was first modeled by Fert and Campbell as if the electron, depending of their spin, were propagating in the material in separated conductive channels with different resistances $r_\uparrow$ for electrons ↑ and $r_\downarrow$ for electrons ↓ (Figure 22b), with $r_\uparrow < r_\downarrow$. This model is generally referred as "two-currents model" and predicts that the resistance of the ferromagnet is expressed as:

$$R = \frac{r_\uparrow r_\downarrow}{r_\uparrow + r_\downarrow} \tag{4}$$

The two-currents model supposes that the spin of an electron is always conserved during the conduction in the ferromagnet and thus does not account for the spin flips that can appear due to thermal fluctuations, or defects in the material. In order to account for this spin-flips, leakage channel with a resistance $r_{\uparrow\downarrow}$ can be added to link the two spin channels. The two-currents model does not account for all the physical phenomena that occur during the diffusion of electrons in a ferromagnet, but gives a satisfying intuitive explanation for magneto-resistive effects.

3.2 MAGNETO-RESISTIVE EFFECTS

Magneto-resistive effects are observed in magnetic junctions, where two ferromagnetic layers are separated by a non magnetic spacer. In these junctions, the current can flow in the plane of the junction: "current in-plane" (CIP). It can also flow perpendicularly to the plane of the junction: current perpendicular to the plane (CPP). The spacer can be metallic, and in the case of these "metallic spin valves", the effect is called "giant magnetoresistance", or it can be a thin insulating layer and in this case of "magnetic tunnel junctions" the effect is named "tunnel magneto resistance".

3.2.1 *Giant magneto-resistance*

Large magneto-resistive effects were first observed in CIP metallic junctions, leading to the discovery of the "giant magneto-resistance" by Fert and Grünberg [121, 122]. In this structure, the relative orientation of the two magnetizations of the ferromagnetic structure implies different resistances. Two extreme configurations can be considered, namely the parallel configuration where the magnetization (Figure 23a) of the two ferromagnetic layers are aligned and the anti-parallel configuration where the magnetizations are in opposite direction (Figure 23b). In the case of the parallel configuration, the electrons that are majority electrons for the first ferromagnetic layer stay majority electrons for the second ferromagnetic layer and respectively, the other electrons are scattered passing through the two ferromagnetic layers.

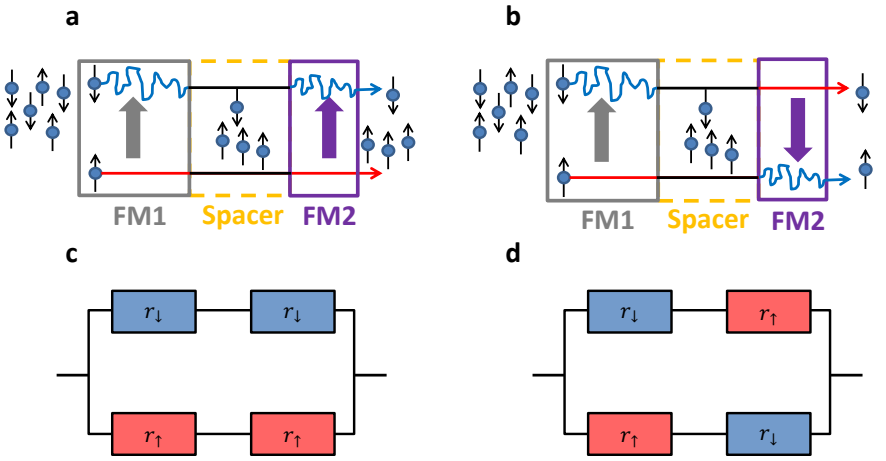

Figure 23: a. Parallel configuration: the electrons scattered at the first ferromagnet are scattered at the second one. b. Anti-parallel configuration: half of the electrons are scattered passing through the first ferromagnet. The other half is scattered passing through the second magnet. c. Equivalent circuit for the parallel configuration. d. Equivalent circuit for the anti-parallel configuration.

Following the two-currents model (Figure 23c), R_P, the resistance of the metallic spin valve in this configuration is given by:

$$R_P = \frac{2r_\uparrow r_\downarrow}{r_\uparrow + r_\downarrow} \tag{5}$$

In the case of the anti-parallel configuration (Figure 23b), the electrons that are scattered at the first ferromagnet layer are diffused at the second. The others are diffused at the first ferromagnetic layer and scattered at the second. The equivalent circuit in the two-currents model is represented in Figure 23d. R_{AP}, the resistance of the metallic spin valve in the anti-parallel configuration is thus:

$$R_{AP} = \frac{r_\uparrow + r_\downarrow}{2} \tag{6}$$

The resistance of the anti-parallel configuration is larger than the resistance of the parallel configuration ($R_{AP} > R_P$) and the relative difference between these two different configurations resistance is computed with the GMR ratio:

$$GMR = \frac{R_{AP} - R_P}{R_P} = \frac{(r_\downarrow - r_\uparrow)^2}{4r_\downarrow r_\uparrow} \tag{7}$$

In an intermediate case, when the relative magnetization is in between these two extreme configurations, the resistance of the spin valve depends on the relative angle θ between the magnetization of the two ferromagnetic layers and can be expressed as follows [123, 124]:

$$R = R_P(1 + \frac{GMR}{2}(1 - \cos(\theta))) \tag{8}$$

Interestingly, measuring the resistance of a spin valve allows accessing the relative angle between the two magnetizations. Thus, the GMR effect has been used in many sensor applications, the most famous and important one being hard-drive reading heads [125]. If the two-currents model gives a good qualitative description of magneto resistive effects, it does not consider the effect of thermal fluctuations or defects in the material. Moreover in a multilayer structure where the current is sent perpendicular to the layers plane (CPP), it does not describe well the physics at the interfaces, where spin accumulation arises. A more predictive model, describing the spin accumulation at interfaces has been developed by Valet and Fert [126]. The first measurements in [121, 122] reported GMR ratio of the order of 1.5% at room temperature. Typically, GMR at room temperature ranges from 1% to 10% [127, 128, 129]. A higher magnetoresistance ratio can be obtained by replacing the metallic spacer with a thin insulating layer, giving rise to the tunnel magnetoresistance.

3.2.2 *Tunnel magneto-resistance*

Tunnel magnetoresistance is known since the seventies [130], but did not attract much interest initially because, due to the fabrication process and chosen materials, the measured magnetoresistance was quite modest even at low temperature. The first high tunnel magnetoresistances were measured in the mid nineties [131, 132]. A more detailed review on tunnel magnetoresistance can be found in [133]. This effect appears in magnetic tunnel junctions, where the spacer is an insulator and thus electrons pass through the barrier due to the tunnel effect. Magneto-resistive effects come in this case from spin-dependent probabilities of tunneling. The first model describing the electronic transport in a magnetic tunnel junction was proposed by Jullière [130]. This model supposes that the spin is conserved

during the tunneling process and that the tunneling probability is proportional to the product of the electron state densities at the Fermi level for the two ferromagnetic layers D^1 and D^2. Thus the conduction of the barrier is also proportional to D^1 and D^2 ($G\ \alpha\ D^1D^2$). This last assumption comes from the previous work of Bardeen [134]. The transport of electrons happens through two different conductance channels depending on the direction of the electron spins.

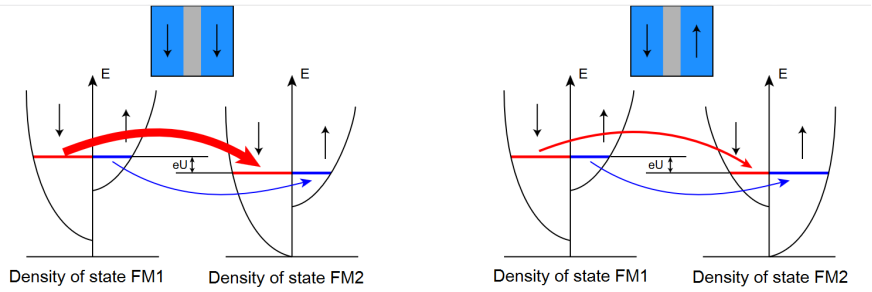

Figure 24: Schematic of principle of the tunneling process in two different configurations: parallel and antiparallel. This schematic illustrates the Jullière model. Larger width arrow signifies a higher probability to tunnel. Figure extracted from [135].

Figure 24 illustrates the two different conduction channels, depending on the magnetic configuration of the barrier. In the parallel configuration, the majority (respectively minority) electrons tunnel to majority state (respectively minority). Thus the conductance G_P of the magnetic tunnel junction is proportional to:

$$G_P\ \alpha\ D^1_\uparrow D^2_\uparrow + D^1_\downarrow D^2_\downarrow \tag{9}$$

Where $D^1_\uparrow$ and $D^1_\downarrow$ (respectively $D^2_\uparrow$ and $D^2_\downarrow$) are the densities of state for majority and minority electrons for the first ferromagnetic layer (respectively second ferromagnetic layer). In the anti-parallel configuration, the majority electrons (respectively minority electrons) in the first layer tunnel to minority states (respectively majority states). In that case the conductance of the junction is defined as:

$$G_{AP}\ \alpha\ D^1_\uparrow D^2_\downarrow + D^1_\downarrow D^2_\uparrow \tag{10}$$

As for the two-currents model, the difference of resistance comes from a difference of state densities at the Fermi level ($D_\uparrow(E_F) \neq D_\downarrow(E_F)$). To

simplify the expression of the tunnel magneto resistance ratio (TMR), which is the normalized difference of resistance between anti-parallel and parallel configuration, Jullière introduces a spin polarization P_i at each layer $i = 1, 2$.

$$P_i = \frac{D^i_\uparrow - D^i_\downarrow}{D^i_\uparrow + D^i_\downarrow} \tag{11}$$

Using the spin polarization of each layer, the TMR is expressed as:

$$TMR = \frac{R_{AP} - R_P}{R_P} = \frac{2P_1P_2}{1 - P_1P_2} \tag{12}$$

Using this expression, Jullière estimates to few tens of percent the TMR of a $Fe/Ge/Co$ junction [130]. The Jullière model does not account for the properties of the barrier. In 1989, Slonczewski introduced a more precise model [136], modeling the insulating layer as a potential barrier. In this model the two electrodes are supposed to be identical with only the angle between the electrode magnetizations which varies. The conductance of the barrier for each spin is computed solving the Schroedinger equations. In this approach, the spin polarization depends on the wave vectors $k_{\uparrow^i}$ and $k_{\downarrow^i}$ for the majority and minority electrons in each ferromagnetic layer i, and interestingly, it depends also on the evanescent propagation of the wave function $\alpha\ e^{-\kappa x}$ in the barrier. The spin polarization of the ferromagnet and insulator couple is expressed as:

$$P_i = \left(\frac{k^i_\uparrow - k^i_\downarrow}{k^i_\uparrow + k^i_\downarrow}\right)\left(\frac{\kappa^2 - k^i_\uparrow k^i_\downarrow}{\kappa^2 + k^i_\uparrow k^i_\downarrow}\right) \tag{13}$$

The expression of the TMR stays the same as in equation 12 by substituting P_i by its new expression. In the case of two magnetizations with relative angle θ, the global conductance of the MTJ can be expressed as:

$$G = G_0(1 + P_1P_2\cos(\theta)) \tag{14}$$

Experimentally, the first TMR ratio measured at room temperature with amorphous aluminum oxide barrier were around 70% [131, 132, 137]. The Slonczewski model is more precise than the Jullière model,

but still fails to describe the influence of the crystallography of the barrier. Using more advanced ab initio simulations, Butler et al [138] predict extremely high TMR ratio (1600 %) for Fe electrodes sputtered on MgO tunnel barriers. In such structures, the Bloch states of the ferromagnetic material hybridize with the evanescent states of the MgO tunnel barrier having the same symmetry. This effect allows filtering spins and increase the spin polarization. This work motivated to fabricate crystalline MgO tunnel barriers, where TMR over 100% were reported [139, 140]. As for GMR, TMR effect is used for sensing and is also intensively studied to read the information stored in new MRAM memories.

3.3 SPIN TRANSFER TORQUE

The second physical effect important for the understanding of the STO is the spin transfer torque. This effect drives the auto-oscillation of spin-torque oscillator. The spin torque effect was first predicted by Berger in 1978. Following its prediction, first spin torque observations were made in [141, 142] to move domain walls, but the intensity required, in the large films they used, was over tens of ampers, limiting the applications. This effect requires indeed high current density and thus the development of nano-technology induced a regain of interest. Spin-transfer torque was relaunched in the late nineties when Berger [143] and Slonczewski [144] predicted that this effect could be strong enough to reorientate the magnetization of a ferromagnetic layer. A more detailed review on spin-transfer torque can be found in [118, 145]. This effect comes from the conservation of the spin momentum. As it has been seen in the previous section, when a current passes through a ferromagnet, the current aligns its spins with the local magnetization (spin polarization). If a spin polarized current is sent to another ferromagnetic layer, it aligns its spins with this magnetization (Figure 25a), so it loses its component which is transverse to the local magnetization (Figure 25b). By conservation of the spin momentum, this component is transfered to the magnetization. This transfer of spin momentum can be interpreted as a torque which is applied to the local magnetization. For high enough current densities, this torque can change the magnetization direction, and thus this effect can be used to manipulate magnetizations.

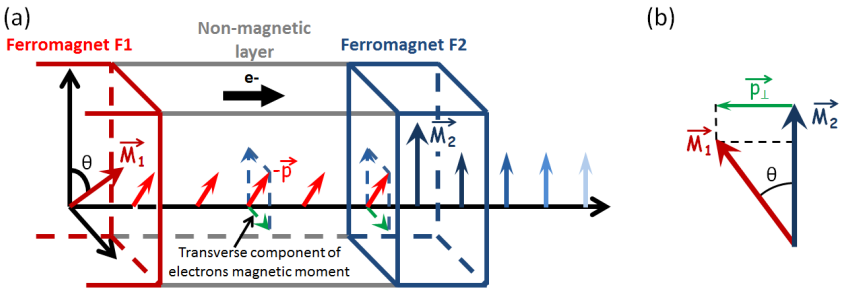

Figure 25: a) Principle of spin-transfer torque in a spin-valve multi-layer structure. b) The transverse component of the spin-polarized current is transferred to the local magnetization of the second ferromagnet. Figure extracted from Lebrun thesis [120].

From the simple representation of Figure 25, the spin torque $\vec{\Gamma}$ is expected to be in the opposite direction of the transverse component $\vec{p_\perp}$: $\vec{\Gamma} \ \alpha \ \vec{p_\perp} = \vec{M}_2 \times (\vec{M}_2 \times \vec{M}_1)$. In reality, due to spin quantum rotation effects, non-colinear components to $\vec{p_\perp}$ arise (field like torque). In order to describe the different components of the spin torque, the effects at the interface between the non-magnetic material and the ferromagnet should be considered [146, 147].

Let's consider an interface between a non-magnetic material (NM) and a ferromagnet, with an incident spin current $\vec{I}_{inc}$ (Figure 26a). The ferromagnet has a magnetization direction $\vec{m}$ and the incident spin current has a polarization direction $\vec{p}$. At the interface, part of this spin current is transmitted ($\vec{I}_{trans}$) and part is reflected ($\vec{I}_{ref}$). The spin torque comes from the fact that the spin current is not conserved at the interface ($\vec{I}_{inc} + \vec{I}_{ref} \neq \vec{I}_{trans}$) and thus this missing part $\vec{\Gamma} = \vec{I}_{inc} + \vec{I}_{ref} - \vec{I}_{trans}$ is fully transmitted as a torque on the magnetization of the ferromagnetic layer. This assumption neglects other dissipations of the spin such as excitation of short-wavelength magnons or transfer to the atomic lattice [118].

Since the component of $\vec{\Gamma}$ parallel to the ferromagnet magnetization has no effect on this last one, we only consider the transverse component of the torque $\vec{\Gamma}^\perp = \vec{I}^\perp_{inc} + \vec{I}^\perp_{ref} - \vec{I}^\perp_{trans}$. The component $\vec{I}^\perp_{inc}$ correspond to $\vec{p_\perp}$ seen earlier. Passing through the interface, the spin of the electrons precesses around the local magnetization (spin precession phenomena) with a period $\frac{1}{k_\uparrow + k_\downarrow}$ (Figure 26). Since the

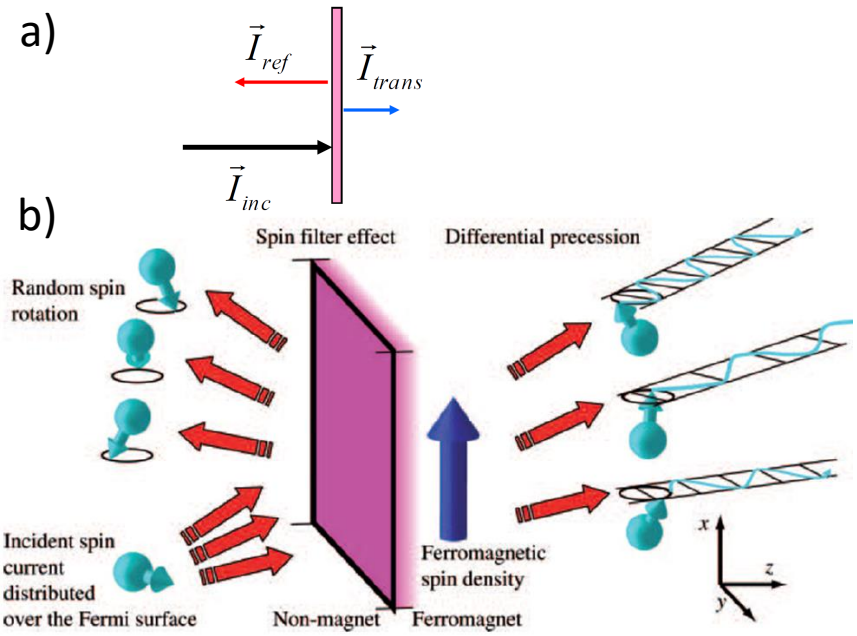

Figure 26: a) Principle of reflection and transmission phenomena of an incident spin-polarized current at an NM/M interface. b) Schematic of mechanisms occurring at the interface. In the lower left corner, the incident spin polarized current is represented with electrons having the same spin state which is transverse to the direction of the magnetization of the ferromagnetic layer. These electrons come with different random incident directions (here represented as having three different incident directions). In the top left corner, the reflected spins are distributed over many directions. In the right corner, the transmitted electron spins are precessing as a function of their distance from the interface. Figure extracted from Stiles and Miltat [145].

electrons in the ferromagnet follow different paths, their precessions are quickly decorrelated (typically after 1 nm in transition metal [148]) and thus we can consider that in average $< \vec{I}^{\perp}_{trans} >= 0$ and neglect this contribution.

The reflected electrons are submitted to spin rotation which is a quantum phenomena. Depending on the incident angle of the electron and on the spin-dependent amplitude of reflection, its spin rotates of a certain angle (Figure 26b for more details see [145]). In a metallic

spacer, the electrons have no favored incident angle and thus the component due to spin rotation is zero in average. For the case of the tunnel barrier, the electrons that have the higher probability to tunnel are the ones that have an incident angle orthogonal to the barrier, because they have the shortest path through the barrier. Since the incident electrons propagate in the barrier with the same incident angle, and the same spin direction, when they are reflected, their spin rotates from the same angle and thus $< \vec{I}^{\perp}_{ref} > \neq 0$. In particular, $< \vec{I}^{\perp}_{ref} >$ can give rise to a component orthogonal to $\vec{M}_2$ and $\vec{p}_{\perp}$.

In the case of a tunnel barrier, the spin torque can be expressed as:

$$< \vec{\Gamma} > \approx < \vec{I}^{\perp}_{inc} > + < \vec{I}^{\perp}_{ref} > \tag{15}$$

This torque can be separated in two contributions: one which is parallel to $\vec{p}_{\perp}$, named Slonczewski torque $\vec{\Gamma}_{slonc}$ and one orthogonal to $\vec{p}_{\perp}$ named field-like torque $\vec{\Gamma}_{fl}$.

$$\vec{\Gamma}_{slonc} = \gamma_{slonc} \vec{m} \times (\vec{m} \times \vec{p}) \qquad \vec{\Gamma}_{fl} = \gamma_{fl} (\vec{m} \times \vec{p}) \tag{16}$$

The two spin-torques depend on the applied voltage [149, 150] and of the nature of the junction [151], and at low bias voltage, in asymmetric junctions, we can approximate they have a linear dependency with the applied current J and the magnetization at saturation M_S of the ferromagnet [152, 153].

$$\gamma_{slonc} = a_j J M_S \qquad \gamma_{fl} = b_j J M_S \tag{17}$$

In this equation, a_j and b_j represent respectively the efficiency of the Slonczewski torque and the field like torque, which can be expressed as follows [152, 153].

$$a_j = \frac{|g| \mu_B}{2|e|} \frac{P}{L M_S} \qquad b_j = r_{fl} a_j \tag{18}$$

In these equations, g is the Landé factor, μ_B the Bohr magneton, e the elementary charge. The efficiency is directly proportional to the spin polarization P and inversely proportional to the thickness L of the free layer. The efficiency of the field-like torque is proportional to the one of the Slonczewski torque with a factor r_{fl} which is typically between 0.1 and 0.4 in magnetic tunnel junctions. An efficient spin torque

requires thus high spin-polarization and thin free layer. This torque can be sufficient to modify the magnetization of a ferromagnet. The effect of the spin transfer torque on the magnetization of a ferromagnet is presented in the next section.

3.4 MAGNETIZATION DYNAMICS

Classically, magnetization dynamics is described by the Landau Litschitz Gilbert equation. The first equation was introduced by Landau and Litshitz in 1933 [154], and Gilbert introduced a similar equation in 1955 with a modified damping term, which is now called the Landau Litschitz Gilbert (LLG) equation [155]:

$$\frac{d\vec{m}}{dt} = -\gamma_0 \vec{m} \times \vec{H}_{eff} + \alpha \vec{m} \times \frac{d\vec{m}}{dt} \tag{19}$$

Here $\vec{m}$ is the local magnetization, γ_0 is the gyromagnetic ratio, $\vec{H}_{eff}$ is the effective field, α is the Gilbert damping factor. When the local magnetization $\vec{m}$ of a material is away from equilibrium, it gyrates around the effective field H_{eff} (first term on the right of the equation). The effective field is determined by the applied external field, by the anisotropy field generated by the crystallographic nature of the material and by the dipolar field generated by the magnetization in other spatial positions. The magnetization gyration is damped (second term of the equation) and $\vec{m}$ tends to align with $\vec{H}_{eff}$. The damping factor α introduced by Gilbert is phenomenological. It can be interpreted as a frictional factor, and its value varies from a material to an other, but is generally of the order of 10^{-2} ($\alpha_{FeB} \approx$0.007-0.008, $\alpha_{NiFe} \approx 0.01$ [156]).

The spin torque contribution can be directly added as a supplementary term to the LLG equation:

$$\frac{d\vec{m}}{dt} = -\gamma_0 \vec{m} \times \vec{H}_{eff} + \alpha \vec{m} \times \frac{d\vec{m}}{dt} - \gamma_{slonc} \vec{m} \times (\vec{m} \times \vec{p}) - \gamma_{fl} (\vec{m} \times \vec{p}) \tag{20}$$

The field like torque can be accounted as a new field contribution $\vec{H}_{fl} = \frac{\gamma_{fl}}{\gamma_0} \vec{p}$ to the effective field $\vec{H}_{eff}$ [157], simplifying the equation 20 as follows:

$$\frac{d\vec{m}}{dt} = -\gamma_0 \vec{m} \times \vec{H}_{eff} + \alpha \vec{m} \times \frac{d\vec{m}}{dt} - \gamma_{slonc} \vec{m} \times (\vec{m} \times \vec{p}) \tag{21}$$

To illustrate the effect of the Slonczewski torque, we will suppose that the spin-polarization is in the same direction as the effective field: $\vec{p} = \frac{1}{\|\vec{H}_{eff}\|}\vec{H}_{eff}$. By replacing in equation 21 $\frac{d\vec{m}}{dt}$ by the right side of equation 21, and by neglecting the term proportional to α^2 (which is approximatively 10^{-4}), equation 21 is rewritten [157]:

$$-\frac{1}{\gamma_0}\frac{d\vec{m}}{dt} = \vec{m} \times \vec{H}_{eff} + \widetilde{\alpha}\vec{m} \times (\vec{m} \times \vec{p}) \tag{22}$$

with $\widetilde{\alpha}$ defined as:

$$\widetilde{\alpha} = \alpha + \frac{\gamma_{slonc}}{\gamma_0\left\|\vec{H}_{eff}\right\|} = \alpha + \frac{a_j M_S}{\gamma_0\left\|\vec{H}_{eff}\right\|}J \tag{23}$$

As can be seen in equation 23, the Slonczewski torque acts as an additional damping term. For positive current J, it reinforces the damping. Interestingly for negative current J, it acts as a negative damping (see Figure 27) and for high enough current (typically 10^{11} $A.m^{-3}$), it can compensate the damping. The different torques acting on the magnetization are schemed in Figure 27. Depending on the

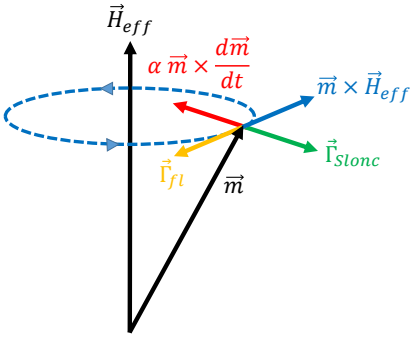

Figure 27: Schematic showing the torques acting on the local magnetization in presence of an effective magnetic field. The red arrow represents the damping which tries to bring back the magnetization along the effective field. The blue arrow corresponds to the magnetization precession around the effective field. The green and orange arrows correspond respectively to the Slonczewski and field-like torques.

intensity of the current J and of the potential barrier, two scenarios can happen for the magnetization. For strong enough current, the

magnetization can reverse. This effect is used to write bits of information in the last generation of MRAM. If the spin torque is big enough to compensate the damping, but not enough to reverse the magnetization, it can give rise to sustained gyrations of the magnetization which is the phenomenon leveraged to generate auto-oscillation in spin-torque nano-oscillators.

4

SPIN-TORQUE OSCILLATORS PRINCIPLES AND BENEFITS FOR NEURONS

In this chapter, we explain the working principle of spin torque oscillators, we do a brief state of the art of these oscillators and we highlight why they are good candidates to build brain-inspired chips.

4.1 WORKING PRINCIPLE

As mentioned in the previous chapter, spin-torque oscillators leverage both magneto resistive effects and spin-torque effects. Figure 28 illustrates the working principle of a spin torque oscillator.

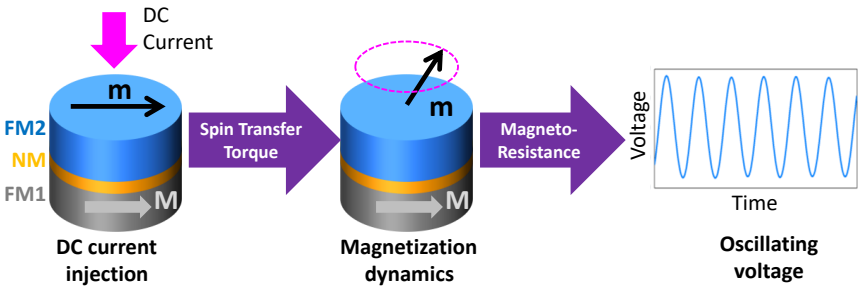

Figure 28: Principle of a spin torque oscillator.

These oscillators are made of two ferromagnetic layers FM1 and FM2, with respectively magnetizations $\vec{M}$ and $\vec{m}$, separated by a non magnetic spacer NM. When a d.c. current is injected, the current is spin polarized when passing through FM1 which is called the reference layer. The spin polarized current passes through NM and then applies a spin transfer torque on $\vec{m}$, the magnetization of FM2 called the free layer. For sufficiently high current density (typically 10^7 $A.cm^{-2}$), this torque can destabilize $\vec{m}$, and induce sustained gyrations of the magnetization. It results in a variation of the relative angle between $\vec{M}$ and $\vec{m}$, and thus by magneto-resistive effects, the resistance of the spin valve varies periodically. This results in an oscillating voltage

across the junction. Interestingly these oscillators are non-linear and can couple and synchronize together through different mechanisms such as spin waves [158, 159], dipolar coupling [160] and electrical coupling [161].

Since such phenomena require high current density, it is observed only in small area devices (typically smaller than 100 nm in diameter). The three main spin-torque oscillator geometries are namely the nano-pillar, the nano-contact and hybrid geometries (Figure 29). These three architectures ensure a sufficient current density in the free layer to destabilize its magnetization. Depending on the geometry chosen, different fabrication process are used. The nano-pillars (which are typically 100-500 nm large) can be fabricated by electron beam lithography and ion-milling [162] or electrodeposition [163] or other techniques [164]. In nano-contacts, the contact can be a sharp mechanical tip which can be downscaled to 10 nm [165], or it can be fabricated using lithography techniques [166, 167]. In this last case, the contact is larger (typically 100 nm). The first evidences of oscillations driven by

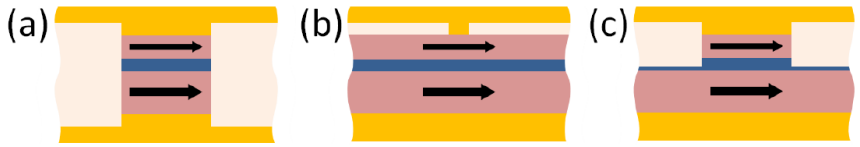

Figure 29: Different geometries for spin-torque oscillators: (a) Nano-pillar (b) Nano-contact (c) Hybrid approach. Extracted from [168].

spin-torque were found by Tsoi in nano-contact geometry [165]. The first direct experimental measurement of oscillations were performed by Kiselev et al [169] in nano-pillar geometry and by Rippard et al [170**?**] in nano-point contact geometry. The important features of the spin-torque oscillator that are being optimized are notably their emission power P, their frequency f, their linewidth Δf and their quality factor $Q = f/\Delta f$. Thus the design of the spin-torque oscillator through the dimensions and the choice of the material for the free layers are crucial to optimize its performance.

It is important to note that the emission power of spin torque oscillators is related to the magneto-resistance ratio and thus depending on the type of junction whether it is a metallic spin valve or a magnetic tunnel junction, the range of emission power varies importantly.

Indeed in metallic junctions, the GMR ratio is typically of 1%-10%. Thus the emission power of such devices is of the order of 1 pW [171] to 25 nW [172]. On the other hand, MgO magnetic tunnel junctions can exhibit TMR superior to 100%, and with such device emission powers over 10 μW were obtained [173].

As mentioned earlier, FM1 and FM2 play different roles and thus their features are different. FM1 the reference layer ensures the spin-polarization of the d.c. current. Thus for an efficient polarization, its magnetization should be as stable as possible. The first approach used for spin-torque oscillators was to use relatively thick layers, so the torque in FM1 is sufficiently low to not disturb its magnetization. More recently in nano-pillars, new approach used a synthetic antiferromagnet (SAF) for the reference layer. FM1 is a stack antiferromagnet/ferromagnet/non-magnetic metal/ferromagnet to obtain a pinned magnetization of the last ferromagnetic layers of the SAF. The first ferromagnet of the SAF couples with the anti-ferromagnet through exchange bias interaction. The two ferromagnetic layers separated by the thin non magnetic metal couple together through the RKKY (Ruderman-Kittel-Kasuya-Yosida [174]) interaction which is chosen so that their magnetizations tend to point in opposite directions. It results that last ferromagnetic layer magnetization is strongly pinned and that the global remanent field of the synthetic antiferromagnetic is weak. The STOs used in these thesis have such reference layer based on a SAF.

The free layer FM2, in the contrary, should have a magnetization which can be easily destabilized. Generally material with a low Gilbert damping such as NiFe or FeB are used. Moreover this layer is generally thin (typically few nm thick). The noise of the oscillator-emission is directly related to the motion of the free layer magnetization. If the free layer is too thin, it becomes more sensitive to thermal fluctuations which is not desired for most applications. Thus a trade-off should be made between a stable magnetization, and an easily destabilized magnetization. Initially the first free layers were chosen to exhibit a uniform magnetization. When they are excited by a spin-polarized current, non uniform modes can appear such as edge modes and center modes. The cohabitation of several modes [175, 176] is source of noise because it induces jumps between modes [177, 178]. In order to create an excitation mode well separated from higher order modes, a vortex

configuration can be used. Using specific aspect ratio in the free layer allows to have a stable vortex configuration, when no field is applied. The vortex based spin-torque oscillator was proposed by Pufall et al in 2007 [179] and Mistral et al [180]. Such configuration has been shown to reduce the phase noise, and in particular linewidths between 0.07 and 1 MHz were obtained in [181]. But when compared with uniform magnetization, these oscillators have a smaller frequency, around hundreds of MHz to 2.2 GHz when uniform magnetization can lead theoretically to 65 GHz [182]. Therefore, even though the vortex oscillators have the smallest linewidth, the highest quality factors were obtained for uniform magnetization [170]. A more detailed state of the art and comparison of the different spin-torque oscillators can be found in the following Table 1. Interestingly, since the first

Ref	Type	Size(nm^2)	P	f(GHz)	Δf(MHz)	Q	MR/R
[183]	NP(GMR)	9100	89 pW	5-10	-	-	1%,8Ω
[184]	NP(GMR)	1300	20pW	11-12	3.2	-	0.4%,13Ω
[185]	NP(GMR)	5000	0.1nW	11-12	3.8	3150	1%,3Ω
[186]	NP(GMR)	11700	5nW	10-11	10	1124	3%,9Ω
[187]	NP(GMR)	6600	1nW	3.5-4	-	266	12%,8Ω
[188]	NC(GMR)	1250	87pW	5-40	58	800	2%,4Ω
[170]	NC(GMR)	1250	87pW	9.7-34.4	1.89	18000	1%,15Ω
[189]	NC(GMR)	1250	70pW	10-46	4.5	7300	0.4%,6Ω
[172]	NC(GMR)	15000	25nW	0.8-2	3	-	10%,10Ω
[171]	V(GMR)	9500	0.8pW	0.9-2.2	0.3	4000	1%,19Ω
[190]	NP(MTJ)	49000	20nW	4-7	21	238	48%,16Ω
[191]	NP(MTJ)	5000	25nW	4-10	26	1000	100%,3kΩ
[192]	NP(MTJ)	45000	142nW	3-12	20	-	70%,43Ω
[193]	NP(MTJ)	13000	550nW	4-7	47	-	66%,143Ω
[194]	NP(MTJ)	31000	200nW	2.6-2.8	80	35	88%,4kΩ
[195]	NP(MTJ)	11000	500nW	2-6.3	46.6	135	66%,2kΩ
[196]	NC(MTJ)	7500	2.4μW	2-5	12	350	46%,38Ω
[197]	NC(MTJ)	7500	63nW	2.5-15	3.4	3200	38%,55Ω
[198]	V(MTJ)	23000	5μW	0.4-0.9	1.1	718	14%,57Ω
[181]	V(MTJ)	70000	1.4μW	0.5-0.5	0.07	6400	128%,35Ω
[173]	V(MTJ)	82000	10.1μW	0.15-0.3	0.16	2000	190%,57Ω

Table 1: Comparison of the different spin-torque oscillators. Extracted from the thesis of Philippe Talatchian [135].

realization of a spin-torque nano-oscillator, the development of new oscillators has been importantly driven by the improvement of the emission power. Figure 30 shows the evolution of the emission power of spin-torque oscillators versus time. One can notice an exponential

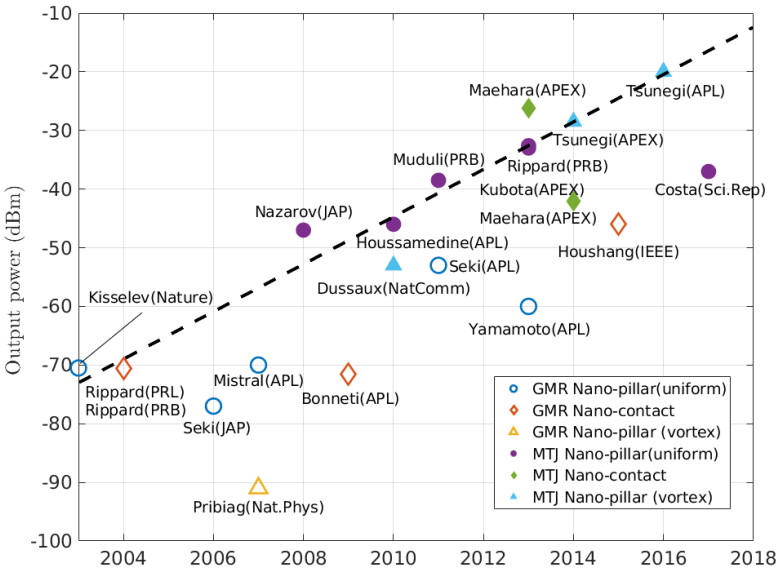

Figure 30: Evolution of spin-torque oscillator emitted power in time. Extracted from the thesis of Philippe Talatchian [135].

progression of the emitted-power in the time with approximatively +3 dBm/year. A last point that should be mentioned is the role of an applied external field. For many spin-torque oscillator designs, an external magnetic field is applied in order to destabilize the free layer magnetization and to ensure that the spin-torque integrated along the whole magnetization trajectory is not zero. This field is generally of the order of 100 mT which may be a drawback for practical applications. Many efforts have been made to obtain oscillations in zero applied magnetic field and different strategies where adopted, depending on the free layer configuration. For uniform magnetization, "wavy shape" of spin-torque [199] and nano-pillar with controlled remanent field direction [200, 201, 202] demonstrate emission at zero external field. For vortex oscillators, zero-field emission was reported for pinned

layers presenting perpendicular magnetization [203, 153] and vortex configuration [204, 205].

The oscillator we use in this thesis corresponds to the highest emission power sample presented in [173]. This oscillator requires an external magnetic field to function but this vortex based sample exhibits high signal over noise ratio and also reduced linewidth (Table 1), which are valuable features for experimental demonstration purpose. In the following section, we present theoretically the behavior of the spin-torque oscillators and the case of uniform and vortex configurations.

4.2 SPIN TORQUE OSCILLATOR NON-LINEAR THEORY

4.2.1 *The non-linear auto-oscillator theory*

Slavin et al describe the dynamics of uniform magnetization using a general non-linear auto-oscillator model [157]. In this model, the whole magnetization is described by a macrospin $\vec{M} = M_x\vec{e_x} + M_y\vec{e_y} + M_z\vec{e_z}$. The norm of $\vec{M}$ is constant $\left\|\vec{M}\right\| = M_0$. Interestingly, the auto-oscillator model extends to describe other types of oscillators, such as Van der Pol oscillators for instance. The oscillator state is described by the complex quantity $c(t)$:

$$c(t) = \sqrt{p(t)}e^{i\theta(t)} \tag{24}$$

where $p(t) = |c|^2$ is the power of the oscillator, and $\theta(t) = arg(c)$ is its phase. The dynamics of the oscillator is governed by the following equation:

$$\frac{dc(t)}{dt} + i\omega(|c|^2)c(t) + (\Gamma_+(|c|^2) - \Gamma_-(|c|^2))c(t) = f(t) \tag{25}$$

Here $\omega(|c|^2)$ is the resonant frequency of the oscillator, $\Gamma_+(|c|^2)$ and $\Gamma_-(|c|^2)$ are respectively the positive and the negative damping of the oscillator and $f(t)$ is the driving force of the oscillator. In the case of a spin torque oscillator, $\Gamma_+(|c|^2)$ is the Gilbert damping and $\Gamma_-(|c|^2)$ comes from spin torque. The non-linearity of this model comes from the dependency of the different parameters ω, Γ_+ and Γ_- with the

power $p = |c|^2$. To describe the dynamics of a macrospin, Slavin et al use for c the quantity:

$$c = \frac{M_x - iM_y}{\sqrt{2M_0(M_0 + M_z)}} \tag{26}$$

and thus the magnetization can be retrieved from c:

$$\vec{M} = M_0(1 - 2|c|^2)\vec{e}_z + M_0\sqrt{1 - |c|^2}\left[(\vec{e}_x + i\vec{e}_y)c + (\vec{e}_x - i\vec{e}_y)c^*\right] \tag{27}$$

The different parameters are derived from the LLG equation with the spin transfer torque term and are expressed as:

$$\omega(|c|^2) = \omega_0 + 2\omega_M|c|^2 \tag{28}$$

where $\omega_0 = \gamma H_0 - 4\pi M_0$ is the ferromagnetic resonance frequency (with H_0 the external field and γ the gyromagnetic ratio), $\omega_M = 4\pi\gamma M_0$ is the coefficient of non-linear frequency shift keying. The positive damping is expressed as:

$$\Gamma_+(|c|^2) = \Gamma_G(1 + Q|c|^2) \tag{29}$$

with $\Gamma_G = \alpha_G\omega_0$ and $Q = (2\omega_M/\omega_0) - 1$.

The negative damping depending on the spin torque is expressed as:

$$\Gamma_- = \sigma_0 I(1 - |c|^2) \tag{30}$$

Here $\sigma_0 = \frac{|g|\mu_B}{2|e|}\frac{P}{LSM_0}$ where S is the surface of the free layer and L its thickness. Oscillations appears only for current I higher than the critical current I_{th}:

$$I_{th} = \frac{\Gamma_G}{\sigma_0} \tag{31}$$

Γ_G depends of the applied magnetic field H_0 and thus by changing the field, one changes the critical current. Introducing the non-dimensional variable I/I_{th}, Slavin et al predict a non-linear behavior of the emitted power with current in the steady state p_0:

$$p_0 = \frac{(I/I_{th}) - 1}{(I/I_{th}) + Q} \tag{32}$$

This formula is obtained using second order expansions with respect with $|c|$ of the parameters Γ_+, Γ_- and ω presented in the first equation. More predictive models can be obtained, adding higher order terms to the different parameters of the equation 25. In particular this first approximation is less valid for high (I/I_{th}). Moreover Slavin et al also developed terms taking into account the sensitivity to noise.

For small perturbations from steady state, the emitted power p relaxes to its steady state p_0 with a characteristic time T_{relax}:

$$T_{relax} = \frac{1}{\Gamma_P} = \frac{1}{\Gamma_G((I/I_{th}) - 1)} \tag{33}$$

Thus the Slavin model predicts a relaxation time which is inversely proportional to the current and which becomes infinite for the critical current. This model predicts the behavior of an oscillator with a uniform magnetization. In our thesis, an oscillator with vortex magnetization is used. We will thus describe in the next subsection this magnetic state, and the model which describes its dynamics. We will link the vortex dynamics with the Slavin auto-oscillator model.

4.2.2 *Non-linear theory in the case of vortex based spin torque oscillators*

The stable magnetic configuration of ferromagnets depends on the dimensions and the shape of the ferromagnet. Indeed the global magnetization distribution results from the competition of the local exchange interaction and the dipolar interaction. The stable magnetization at remanence is the one which minimizes the energy associated to these two interactions. The exchange interaction is acting on the exchange length $L_E = \sqrt{\frac{2A}{\mu_0 M_S^2}}$ with A the stiffness exchange constant and M_S the saturation magnetization of the material ($L_E(NiFe) \approx 5.7$ nm, $L_E(FeB) \approx 11.8$ nm). The dipolar interaction is acting on distances of typically up to 1 μm. Thus the dimensions of the ferromagnet can favor one of these two contributions. The influence of the shape comes from the symmetry it imposes to the magnetization. For example a cylindric geometry ferromagnet favors a cylindrical geometry magnetization. The vortex configuration arises notably in thin cylinders where both interactions are non negligible. Locally the spins should

align and in a more distant range they should be opposite. The vortex magnetization presents two regions (Figure 31): one large where the magnetization is radial and contained in the plane of the cylinder and another one much smaller where the magnetization is out of this plane (vortex core). The size of the vortex core is typically 10 nm. This configuration was predicted in the late nineties [206]. Depending on

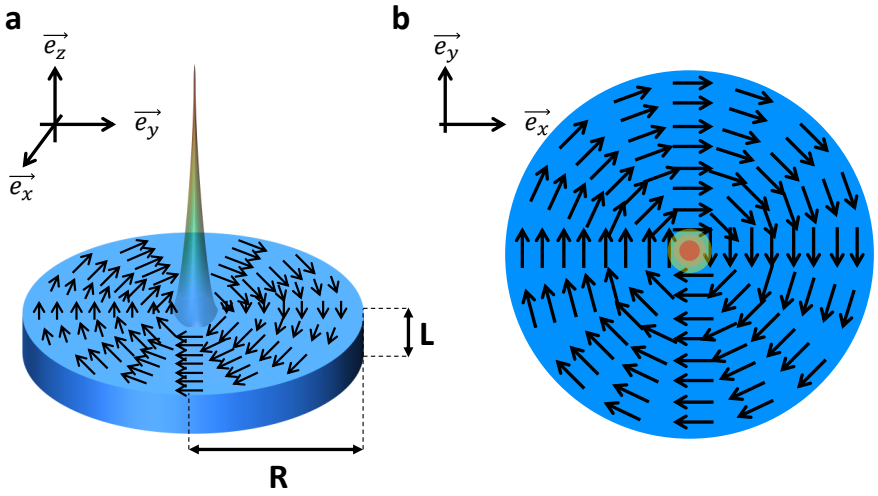

Figure 31: a. Cylinder exhibiting a vortex magnetization. The thickness of the cylinder is called L and its radius R. The vortex magnetization is in the plane of the cylinder except in the core area where the magnetization is out of the plane. b. Schematic of the vortex magnetization in the (x,y) plane. The local magnetization is radial except in the vortex core and has always a fixed value.

the direction the core is pointing towards: $\vec{e}_z$ or $-\vec{e}_z$ (polarity $p = 1$ and $p = -1$) direction and if the in plane magnetization is enrolled in the trigonometric or anti-trigonometric direction (chirality $c = +1$ or $c = -1$), the vortex configuration possesses 4 stable degenerated configurations.

The vortex configuration is stable for particular dimensions (radius R and thickness L) of the cylinder depending on the exchange length L_E of the material (Figure 33). Figure 33 only represents the stable configurations for moderate values of R/L_E and L/L_E. Higher values would lead to multi-domains configurations. Interestingly, spin torque can move the vortex magnetization resulting in a non-zero mean

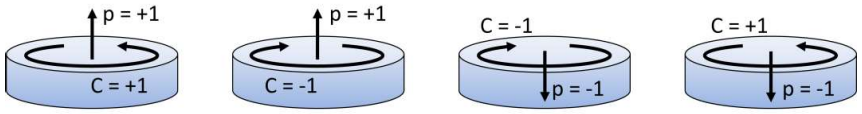

Figure 32: Schematic of the four degenerated configurations of a vortex magnetization. Extracted from Grimaldi [207].

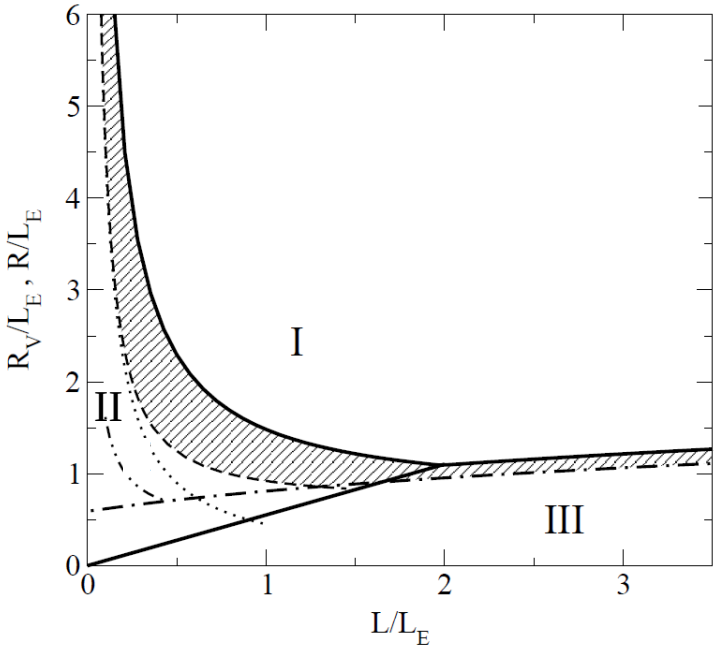

Figure 33: Stability diagram of the three different configurations of the magnetization distribution in a ferromagnetic cylinder having a section radius R and height L. The diagram presents three distinct regions. I: magnetic vortex state, II: magnetized uniformly inplane, III: magnetized uniformly out-of-plane. The dashed region corresponds to a metastable configuration between the three different ground states. Extracted from Metlov et al [208].

magnetization in the plane of (x,y) defined by the cylinder. For small displacements, the vortex can be seen as rigid and thus the magnetization dynamics can be entirely described by the position $\vec{X}$ of the vortex core. Later we will define the vortex core position

in polar $(\vec{e}_r, \vec{e}_\theta)$ coordinates with its radius s and its angle θ. This simplified model is used by Thiele to describe the dynamics of the vortex.

$$\vec{G} \times \frac{d\vec{X}}{dt} - D(\vec{X})\frac{d\vec{X}}{dt} + \vec{F}_{STT}(\vec{X}) - \frac{\partial E}{\partial \vec{X}} = \vec{0} \tag{34}$$

The different forces acting on the vortex core are schemed in Figure 34.

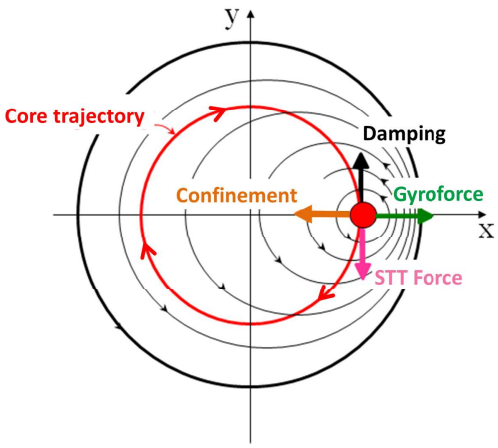

Figure 34: Schematic of the different forces acting on the vortex core. Extracted from Lebrun Thesis [120].

$\vec{G}$ is the gyroforce.

$$\vec{G} = -p\left(2\pi L\frac{M_S^{free}}{\gamma}\right)(1-\cos(\theta_0))\vec{e}_z = -pG\vec{e}_z \tag{35}$$

Where $\theta_0 = \cos^{-1}\left(\frac{H_\perp}{\mu_0 M_S}\right)$ is the free layer magnetization angle, M_S^{free} is the saturation magnetization of the free layer, L is the free layer thickness and p is the vortex polarity.

$$D = D_0(1 + \xi\left(\frac{s}{R}\right)^2) \tag{36}$$

D_0 is expressed as:

$$D_0 = \alpha_G(2\pi L\frac{M_S}{\gamma})\left(\frac{1}{2}\ln\left(\frac{R}{2b}\right) - \frac{1}{8}\right)\sin^2(\theta_0) \tag{37}$$

$b = 2L_E = 2\sqrt{\frac{2A}{\mu_0 M_S^2}}$ and the parameter $\xi \approx 0.6$ in our case. The confinement energy E can be expressed as:

$$E = E_{ms} + E_{oe} \tag{38}$$

where E_{ms} is the magnetostatic energy:

$$E_{ms} = \frac{1}{2}\kappa_{ms}s^2 + \frac{1}{4}\kappa'_{ms}\frac{s^4}{R^2} + O(s^6) \tag{39}$$

with $\kappa_{ms} = \left(\frac{10}{9}\right)\mu_0 M_S^2 \frac{L^2}{R}\sin^2(\theta_0)$ and $\kappa'_{ms} = 0.25\kappa_{ms}$. E_{oe} is the energy associated to the Oersted field confinement:

$$E_{oe} = \frac{1}{2}\kappa_{oe}CJs^2 + \frac{1}{4}\kappa'_{oe}CJ\frac{s^4}{R^2} + O(s^6) \tag{40}$$

Here C is the vortex chirality, J is the electrical current density injected in the nanopillar. The coefficients are expressed as $\kappa_{oe} = 0.85\mu_0 M_S LR\sin(\theta_0)$ and $\kappa'_{oe} = -0.5\kappa_{oe}$. The four coefficients for the energy (κ_{ms}, κ'_{ms}, κ_{oe} and κ'_{oe}) were calculated in [209, 210, 203].

And finally the value of the force exerted by the spin torque $\vec{F}_{stt}$ can be expressed as:

$$F_{STT} = a_j J\vec{e}_\theta \tag{41}$$

with a_j expressed as:

$$a_j = \pi\frac{\hbar P}{2e}\frac{H_\perp}{\mu_0 M_S^{pol}}\sin^2(\theta_0) \tag{42}$$

where P is the spin polarization, $\hbar$ is the Planck constant, e the elementary charge and M_S^{pol} is the saturation magnetization of the polarization layer. From these equations, and defining $\zeta = \frac{\kappa'_{ms}+\kappa'_{oe}J}{\kappa_{ms}+\kappa_{oe}J}$ and $\kappa = \kappa_{ms} + CC_{oe}\kappa_{oe}J$ (C_{oe} is the chirality of the Oersted field), the radius and the phase of the oscillator dynamics are described by the following equation system:

$$\begin{cases} \frac{ds}{dt} + \left(\frac{D\kappa}{G^2}[1 + (\zeta + \xi)s^2] - \frac{a_j Jp}{G}\right)s = 0 \\ \frac{d\theta}{dt}s - p\frac{\kappa}{G}(1 + \zeta q)s = 0 \end{cases} \tag{43}$$

Interestingly, Grimaldi et al [153] show that with $c(t) = s(t)e^{i\theta(t)}$, equation 43 is equivalent to the auto-oscillator equation 25 with:

$$\begin{cases} \Gamma_+(s^2) = \frac{D\kappa}{G^2}[1 + (\zeta + \xi)s^2] \\ \Gamma_-(s^2) = \frac{a_j J}{G} \\ \omega(s^2) = \frac{\kappa}{G}(1 + \zeta s^2) \end{cases} \tag{44}$$

In this case, the critical current is expressed as:

$$I_{th} = \frac{D\kappa\pi R^2}{a_j G} \tag{45}$$

The power of the emission becomes for current over I_{th}:

$$p_0 = \frac{(I/I_{th}) - 1}{\zeta + \xi} \tag{46}$$

So the vortex radius depending on input current evolves typically as:

$$s_0 = \sqrt{\frac{(I/I_{th}) - 1}{\zeta + \xi}} \tag{47}$$

with a square root dependency with the input current. Thus in the auto-oscillation regime, the voltage emitted by the oscillator is expected to be $e_0(t)$:

$$e_0(t) = \lambda s_0 \cos(\omega t) \tag{48}$$

with λ a parameter equal to:

$$\lambda = \frac{I\Delta R_{p-ap}(I)}{2}\beta\sqrt{\left[1 - \left(\frac{H_\perp}{4\pi M_S^{pol}}\right)^2\right]\left[1 - \left(\frac{H_\perp}{4\pi M_S^{free}}\right)^2\right]} \tag{49}$$

where $\Delta R_{p-ap}(I)$ is the difference of resistance between the parallel and the anti parallel state at a given I, M_S^{free} is the free layer saturation magnetization, and β is the conversion factor of the vortex core displacement into magnetization change [209] which is $\beta = \frac{2}{3}$ for s which is less than 60 % of the pillar radius R. Interestingly, the amplitude of the oscillations $\widetilde{V} = \lambda s_0$ evolves like the vortex core

radius and thus depends non-linearly on the applied current I. For vortex motion, the relaxation time of the oscillator should be:

$$T_{relax} = \left[-\frac{D\kappa_{ms}}{G^2} + \left(\frac{a_j}{G\pi R^2} - \frac{D\kappa_{oe}}{G^2 \pi R^2} \right) I \right]^{-1} \tag{50}$$

As for the model of Slavin et al [157] for uniform magnetization, a dependency inversely proportional to the applied current I is expected.

4.3 POTENTIAL APPLICATIONS OF SPIN-TORQUE OSCILLATOR

4.3.1 *RF emitters*

The first straight-forward potential application for spin-torque oscillators is to use them as tunable radio-frequency emitters for instance for telecommunication purpose. Spin torque oscillators are promising to replace the present electronic and CMOS voltage controlled oscillators (VCO) that have a non-negligible size (see for instance section 2.1.1 on CMOS ring oscillator), which is a draw back for embedded systems such as cellphones. Spin-torque oscillators present improvement because of their nanometric size and because they are resilient to radiation (which is particularly important for aeronautic and spatial applications). Unfortunately present spin-torque oscillators do not meet yet the requirements for such systems, because the emitted power should be of the order of 1 mW (instead of 10 μW for the best spin-torque oscillator) and the noise should be of -110 dBm/Hz when best oscillators have a level of noise of -90 dBm/Hz [211]. In order to decrease the level of noise, phase-locked-loop spin-torque oscillators were proposed [212, 213]. In order to both improve the power and decrease the noise, synchronized oscillators are also studied [214].

Spin-torque oscillators are also studied to transmit information through the amplitude variation of the oscillation (similarly to the strategy used for radio signals). This "amplitude shift keying" was demonstrated with a rate of 200 kb/s but potentially it could be improved to 1.5 Gb/s [215]. Encoding the information in the frequency (frequency shift keying) instead of the phase demonstrated higher rates of 400 Mbit/s [216, 217].

Finally spin-torque oscillators are thought to be used in the magnetic memory field. They are studied to read the information at a higher rate [218]. The idea is to use a spin-torque oscillator instead of a GMR or TMR sensor as reading head. The magnetic field that stores the information quickly changes the emission frequency of the spin-torque oscillator. The bit information is extracted from the changes in the emitted frequency of the STO. This would allow reaching a reading rate beyond 3 Gb/s. They are also studied as writing heads [219]. In that case a spin-torque oscillator emits an RF field which helps reverse the magnetization of the memory cell.

4.3.2 *RF detectors*

Interestingly, when a spin-torque oscillator is excited by an RF current, whose frequency is close to the eigen frequency of the STO, the STO generates a d.c. voltage. This effect is called "spin diode" [220]. This effect was proposed to detect the frequency of RF signals [221]. The advantage, when compared with Schottky diodes which are commonly used for frequency detection, is that STO have smaller dimensions and a higher sensitivity [222].

In vortex based STO, the stimulation by a current or a field at the resonance frequency can induce also an expulsion of the vortex core. Interestingly this phenomena improves the signal measured when compared with standard spin-diode, and it can be leveraged for instantaneous frequency detection [221].

Another strategy to improve the spin-diode signal through injection locking for frequency detection was reported in [223]. STOs are thus promising for building compact frequency detectors for instantaneous detection with high sensitivity and a large frequency range.

Spin diode effect can also be leveraged for energy harvesting. A STO stimulated by a RF field can generate a d.c. voltage which could potentially supply a device [224]. This could be particularly promising for low power embedded devices, which cannot be easily energy supplied as for instance medical implants.

4.3.3 *Potential for neuromorphic computing*

As presented in chapter 1, neurons can be seen as non-linear oscillator that compute through their dynamical response and that can couple and synchronize together. Conventional computers have a Von-Neuman architecture which is not well suited for emulating neural network response. Building a brain-inspired architecture requires components whose physics is close enough to neurons for allowing computation. Moreover, solving complex cognitive tasks requires large number of neurons, and thus nanoscale components are required. In this context, STOs are promising candidates since they have nanometric size, and that a single STO is by itself a non-linear auto-oscillator. It is a major difference, for instance when compared with memristors, where auto-oscillations generally require at least an additional capacitance. Similarly to neurons, STOs have a non-linear behavior. In particular, they emit only for current inputs higher than threshold value, which in the case of STOs is the critical current. Moreover the amplitude of the oscillation depends non-linearly on the input current value. Through the relaxation time, the STOs also exhibit a memory which can be leveraged for neuromorphic computing. And for building large arrays, the STOs can couple and interact through different mechanisms. Compared with other nanoscale devices, STOs work at room temperature, and they have a long lifetime, with a relatively low noise and a behavior which does not strongly drift in time. Finally, these oscillators have the same structure as magnetic random access memories, which suggests they could be densely integrated on chips and that they can be interfaced with CMOS circuits.

These promising features motivate theoretical works on the use of STO for brain-inspired computation [225]. At the beginning of this work, there were no experimental demonstrations of brain inspired computing with STOs. The brain computes both using synchronization pattern and transient dynamics. In parallel to my work, brain-inspired computing based on synchronization pattern was also demonstrated using the synchronization of four coupled STOs [226]. This approach is different and complementary to the one adopted in my work (for more informations see [226, 135]). In the work presented in this manuscript, brain-inspired computing is performed by leveraging the transient dynamics of spin-torque oscillators.

Part III

FIRST DEMONSTRATION OF NEUROMORPHIC COMPUTING WITH A SPIN-TORQUE NANO-OSCILLATOR

5

RESERVOIR COMPUTING WITH A SINGLE NON-LINEAR NODE

In this chapter, the computing method called reservoir computing with a single node is presented. This method will be used in chapter 7 to prove neuromorphic computing with spin torque oscillators. The first section of this chapter presents the general principle of reservoir computing. The second section shows how this computation can be adapted to a single non-linear node system using a temporal complexity instead of the more classical spatial complexity of a neural network. Finally, the last section presents a state of the art for single node reservoir computing with hardware systems, showing that this method can be adapted to different hardware.

5.1 RESERVOIR COMPUTING GENERAL PRINCIPLE

Reservoir computing is a kind of recurrent network developed in machine learning to process sequence problems without encountering the classically hard problem of training connections in a recurrent network.

5.1.1 *Reservoir computing architecture*

Reservoir computing was introduced by Maass [27], Jaeger [26, 227] and Steil [228]. Jaeger referred also to this system as "echo state network", Maass as "liquid state machine" and Steil as "backpropagation decorrelation". In all cases, the idea is still to leverage the non linear-dynamics of recurrent neural networks for machine learning [229]. This approach was used successfully to perform recognition tasks. In particular it was used for robot motion control [230, 231], channel equilibrium [227], speech recognition [232, 233, 234, 235], forecasting [236]. This idea was successfully adapted to different hardware because the main requirement is the use of a non linear dynamical system which

can be built from many different materials. A reservoir computing

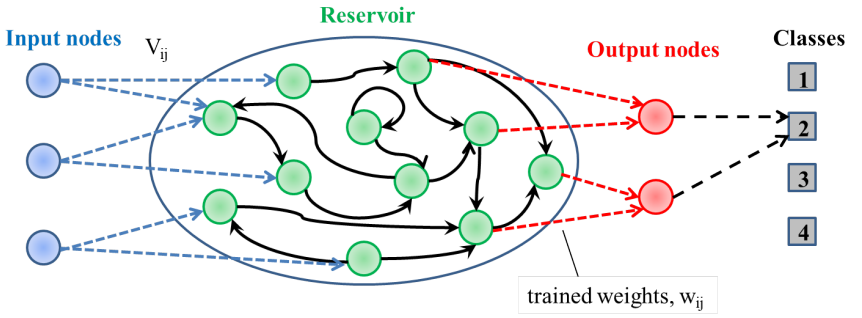

Figure 35: Reservoir computing schematic: the network is composed of three parts. The input is connected with fixed connections to a recurrent network called a reservoir. The connections of the reservoir are also fixed. The reservoir is connected to the output with trained connections. The output allows identifying the class of the input.

system is composed of three distinct parts (Figure 35) which are an input layer, a recurrent network called reservoir and an output layer. The connections between the input layer and the reservoir and inside the reservoir are fixed. Only the connections between the reservoir and the output layer represented in red in Figure 35 are tuned. The input and the output can have various dimensions (for example both the input and the output have a dimension 1 for sine square task, whereas for spoken digit recognition task with cochlear decomposition, the input has a dimension 78 and the output has a dimension 10). In the case of classification problems, the output allows to associate an input to different classes. For speech recognition the different classes could be for instance the different words to identify, whereas the output is a vector of numerical values. Let's call $\mathbf{x}(k)$ the vector containing all different neuron responses from the reservoir $x_i(k)$ at time k.

$$\mathbf{x}(k) = \begin{pmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_n(k) \end{pmatrix} \tag{51}$$

In practice the evolution of the reservoir network is described by the following equation:

$$\mathbf{x}(k) = f_{nl}(W_{in}^{res} u(k) + W_{res}^{res} \mathbf{x}(k-1)) \tag{52}$$

Where W_{in}^{res} is the matrix containing the connection weights between the input $u(k)$ and the reservoir (represented with blue arrows in Figure 35) and W_{res}^{res} is the matrix containing the connection weights inside the reservoir (represented with black arrows in Figure 35). f_{nl} refers to the non linear transformation operated by the neurons of the reservoir (each green circle in Figure 35 operates a non-linear transformation). It is important to note that these three parameters W_{in}^{res}, W_{res}^{res} and f_{nl} are fixed. The response of the reservoir $\mathbf{x}(k)$ depends on the input sequence it receives. Because of the recurrences in the reservoir, $\mathbf{x}(k)$ depends of $u(k)$, but also indirectly of previous input values (through the $x(k-1)$ term).

Then the output $y(k)$ is obtained by combining together the neuron outputs $x_i(k)$. In practice this step is achieved by a linear combination:

$$y(k) = W_{res}^{out} \mathbf{x}(k) + W_{bias} \tag{53}$$

W_{res}^{out} is the matrix containing the connection weights between the reservoir and the output (represented in red in Figure 35) and W_{bias} is a bias vector containing constants (acting like an offset). By defining s and W such as:

$$s(k) = \begin{pmatrix} 1 \\ \mathbf{x}(k) \end{pmatrix} \quad W = \begin{pmatrix} W_{bias} & W_{res}^{out} \end{pmatrix} \tag{54}$$

The equation for the output can be easily rewritten such as:

$$y(k) = W s(k) \tag{55}$$

This simplified notation will be used also in later sections because it simplifies the way learning is written.
So in the end the computation performs well if a linear regression on the responses of the reservoir neurons can approximate well the correct output. The goal of the reservoir is not to solve directly the problem but to turn the initial problem into a new one that is linearly solvable. The next subsection presents the important properties that the reservoir should have to fulfill this purpose.

5.1.2 *Reservoir key properties*

As mentioned in the previous subsection, the goal of the reservoir is not to solve directly the problem but to map the initial problem in a space where it is easier to solve. The main idea is that projected in higher dimensional state, the problem is more likely to become linearly separable [237]. Thus after this projection, the problem can be easily solved by the linear combination operated by the connection layer between the reservoir and the output layer. To perform properly this task, the reservoir must respect three key properties which are the separability, the approximation and the fading memory [238, 239].

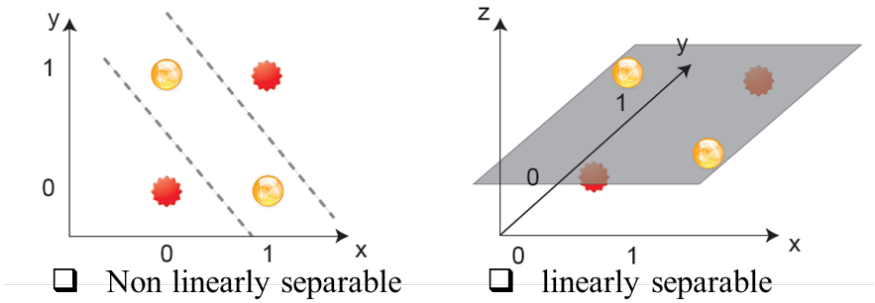

Figure 36: Illustration of separation property (extracted from Appeltant et al [239]): here the separability is illustrated on a XOR task. In two dimensions the two classes can not be separated by a single hyperplane (a straight line). Projected in three dimensions the problem can be separated by a hyperplane (a plane).

The separation property refers to the ability to return different reservoir responses when different inputs are sent. This property is important to obtain a linearly separable problem. It can be achieved through the high number of neurons in the reservoir and their interconnections. The link between this property and a higher dimension mapping is illustrated in Figure 36 in the case of a XOR task. As illustrated in this simple case, the two classes of objects (red stars and yellow spheres) cannot be separated by a unique hyperplane (in 2 dimensions, a line). By adding a third dimension, the problem becomes linearly separable: the two classes of objects can be separated by a plane (the hyperplane in 3 dimensions). The higher dimensional state corresponding to an

input $u(k)$ is the vector $\mathbf{x}(k)$ containing the response to this input of all the reservoir neurons $x_i(k)$. Therefore each response of a reservoir neuron $x_i(k)$ can be seen as a coordinate of this reservoir state. In order to perform well, the number of neurons in the reservoir is generally higher than the dimension of the input. Moreover to perform the higher dimensional mapping, the neurons must have a non-linear behavior, so the neuron responses are independant variables. Leveraging higher dimension mapping is used by other machine learning algorithms such as support vector machine and kernel methods. The particularity of reservoir computing is the interconnection between non-linear nodes allowing successive non-linear transformations of the input.

The approximation property is that similar inputs give similar responses. To fulfill this requirement, the reservoir should be resilient to noise, because otherwise a same input with some small noise could give very different responses and be classified differently. Thus the behavior of the reservoir should not be chaotic. The approximation property can be seen as limiting the separation property, because, according to approximation property, inputs will not be separated if the difference between them is too small. Thus some trade off should be made between these two properties.

Finally the reservoir must have a fading memory, which is a short range memory of recent events. This memory comes from the recurrence in the reservoir (internal loops formed by the connexions between neurons where information is stored) [240]. This property is important to process sequences of data where memory is required.

The separation and fading memory requirements are task dependent. Typically a more complex problem requires a reservoir with more neurons and connections. Fading memory depends also how far in the past the reservoir should remember in order to classify inputs. When these requirement are met, the reservoir maps the input in a higher dimensional state where the useful features for classification appear. The connection between the reservoir and the output can be interpreted as readout weights which select in the complex response of the reservoir the useful features for recognition.

5.1.3 *Learning procedure*

Reservoir computing belongs to the category of supervised learning algorithms. The general principle is thus the same as other algorithms (such as support vector machines, deep learning or long short term memory). The neural network goes through two phases : first a training phase (or learning) where the neural network " learns " to solve the problem and a test phase (or inference) where the neural network applied what was learned during the training phase to solve new problems. The system has some internal parameters that are tuned during the training phase (in the case of reservoir computing it is the connection weights W_{res}^{out} and the bias weights W_{bias}). During training some examples are presented to the neural network. For these examples the correct output is known. The learning procedure consists in finding a set of parameters that minimize the error between the correct output and the actual output of the neural network. A classical error to minimize is the L_2 norm of the difference between the obtained output and the desired output. This error normalized by the number of examples is called root mean square error [241]. Let's call $\widetilde{Y}$ the correct output for all the training examples and Y the actual output of the reservoir computing network. The root-mean-square error for training examples is expressed as follows:

$$RMS = \frac{1}{N_{train}} \|Y - \widetilde{Y}\|_2 = \frac{1}{N_{train}} \sqrt{\sum\nolimits_{k \in training} (y(k) - \widetilde{y}(k))^2} \tag{56}$$

The optimal weights are found such as W_{res}^{out} and W_{bias} minimize the error for all the training examples. Using the simplified notation seen in previous section, we must find W which minimizes the error. Let's define S the reservoir state (responses of the reservoir neurons plus a 1) for all the training examples.

$$S = \Big(s(k)\Big)_{k \in training} \tag{57}$$

The optimal W for the training examples is found doing the following operation:

$$W = \widetilde{Y} S^{\dagger}. \tag{58}$$

† represents the Moore-Penrose pseudo inverse [242]. After finding the optimal W for training examples, the learned parameters are tested on new examples (belonging to a test set). The performances are also measured using the root-mean-square error. If the error is low for the training examples and high for the test examples, it means that the neural network "overfits" to the training examples.
Overfitting is common problem in machine learning. This problem appears when the model (here the neural network with tuned connections) processes well the training examples, finding features in the training examples which do not describe well examples during the test (so it is not generalized). This bad generalization may be due to a bad training set (the distribution in the training set is not the same as for the test set) or due to a too complex model (here it would be too many neurons in the reservoir compared to the number of training examples). In this second case neural networks have enough parameters to extract features from the noise of the data. This overfitting to the training example will be translated to very large values of weights W (the weights somehow amplify a small regularity in the noise of the training examples). In order to avoid this problem, Tikhonov regularization (also referred as ridge regression) is used [243]. With this regularization, the error to minimize is defined as follows:

$$\|Y - \widetilde{Y}\|_2 + \lambda \|W\|_2 \tag{59}$$

Here λ is a small positive factor. The addition of a term $\lambda\|W\|_2$ in the error penalizes the large values of W, so optimal weights would be smaller avoiding to overfit by detecting features in the noise of the training examples. Also it is important to note that λ is a new hyperparameter that needs to be tuned. Another advantage of Tikhonov regularization is that an optimal set of weights always exists and is unique ensuring the convergence of the training algorithm.

5.2 SINGLE NON-LINEAR NODE RESERVOIR COMPUTING

In this section we explain how a single node can emulate the behavior of a reservoir. This method simplifies drastically the experimental implementation of reservoir computing and will be used in the next

chapters 6 and 7 to demonstrate neuromorphic computing with spin-torque nano-oscillators.

5.2.1 *General idea*

The general idea of single node reservoir computing [239] is to replace the spatio-temporal response of reservoir computing classical networks by the purely temporal response of a single node. In the later case, the different temporal states of the single node are equivalent to the neuron outputs of a spatial network. Later we will call these temporal states "temporal neuron outputs".
The single node plays the role of several neurons (we called them "temporal neurons") that have varying responses in time. The initial input needs to be preprocessed, so that the response of the single node is analog to the whole neural network output.
The principle of single node reservoir computing is illustrated in Figure 37. To emulate in time the connection between the input and the reservoir, the sequence to process $u(k)$ is preprocessed through convolution by a mask thus forming a preprocessed input $J(t)$ (this point will be explained in section 5.2.2). This preprocessed input $J(t)$ induces variation of the single node temporal states. These temporal states are combined to reconstruct the output.

The preprocessed input should be specially designed to generate temporal states of the single node that are equivalent to neuron responses. The next subsection explains how to build the preprocessed input $J(t)$.

5.2.2 *Preprocessing of the input*

To behave as a reservoir, the non-linear node needs to be driven by the preprocessed input $J(t)$ it receives. A single node response $x(t)$ should have a behavior similar to several neurons outputs x_i. Thus the preprocessed input should impose that the single node generates sequentially the output of the different neurons responding at different times. To achieve this time multiplexing of the problem, the input needs to be discretized. The input to process becomes a discrete sequence $u(k)$ where the index k refers to the discrete time.

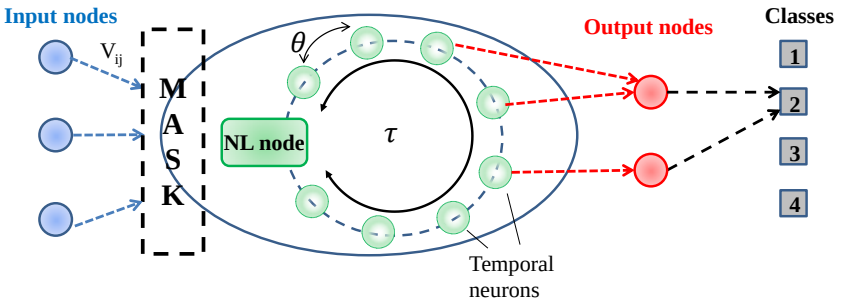

Figure 37: Reservoir computing with a single non-linear node and time multiplexing: the input is sent on a single node after being convoluted by a mask. The single node plays one after the other every θ time step the different temporal neurons. The time it takes to play all the temporal neurons one after the other is called τ. The different temporal states of the single node, which correspond to the temporal neuron outputs, are combined linearly to obtain the output and identify the class of the input.

The preprocessing is illustrated in Figure 38 for a one dimensional input $u(k)$. Figure 38a shows the classical case of a discrete input sent to several neurons of a network, which is the situation we want to reproduce with one single node and time multiplexing. For simplicity, the input $u(k)$ is a sequence of three values and only three neurons are represented. Each neuron receives the input rescaled by the connection weight $W_{in}^{res}(i)$ (so the neuron i receives the input $W_{in}^{res}(i)u(k)$) and gives a response $x_i(k)$).
To emulate the same behavior with a single node, the input $u(k)$ is preprocessed (Figure 38b). The preprocessed input $J(t)$ (magenta curve) takes sequentially the value of the different signals received by the neurons, which are the values $W_{in}^{res}(i)u(k)$ (sequence of red, cyan and green rectangles in Figure 38b). Every time θ, $J(t)$ takes the input value of an other neuron. In particular, in Figure 38b, $J(t)$ takes firstly the value $W_{in}^{res}(1)u(1)$ (red rectangle), which is the signal received by the neuron 1 at time $k = 1$, and after a time θ, it takes the value $W_{in}^{res}(2)u(1)$ (blue rectangle), which is the signal received by the neuron 2. So $J(t)$ takes the value of the input signals for all the n neurons at a given time step k in the time interval $\tau = n\theta$ (in Figure 38, $n = 3$). After τ, $J(t)$ takes the input value of the

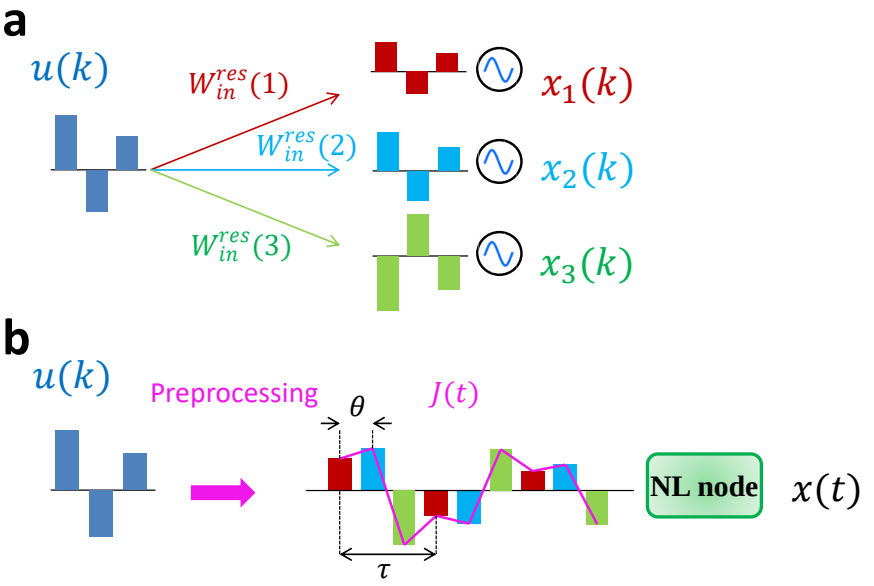

Figure 38: a. Schematic of the injection of an input in a spatial network. The input $u(k)$ is first multiplied by a weight w_i (red, cyan and green rectangles) before being injected in the neuron i which produces an output $x_i(k)$. In this example the input $u(k)$ is a one dimensional sequence of three values and only three neurons are represented. b. Time multiplexing of the problem: the input $u(k)$ is preprocessed forming a continuous input (represented in magenta) $J(t)$. $J(t)$ takes sequentially the values of the different neuron inputs $w_i u(k)$ (red, cyan and green rectangles). Every time θ, $J(t)$ takes the input value of an other neuron. Every time τ, $J(t)$ takes the input value of the same neuron but at another time step k. This continuous input $J(t)$ is injected in the single node, which produces a continuous response $x(t)$.

same neuron but at the next time step $k+1$ (in Figure 38b, the two consecutive red rectangles are separated by a time τ). If the dimension of the input u is 1, the value of the input $J(t)$ is given by the following equation:

$$J((k-1)\tau + i\theta) = W_{in}^{res}(i)u(k) \tag{60}$$

This equation imposes the value of $J(t)$ only at times t that are multiples of θ. To have $J(t)$ continuous in time, we connect the values imposed by equation 60 with straight lines (as represented in Figure 38b). In practice to obtain $J(t)$, $u(k)$ is convoluted with a

sequence which takes the $W_{in}^{res}(i)$ values every θ. This sequence is generally referred as a mask in the literature. If the dimension of the input is greater than one, the previous equation 60 can be extended:

$$J((k-1)\tau + i\theta) = \sum_{j=1}^{j=dim(u)} W_{in}^{res}(i,j)u(k) \tag{61}$$

The preprocessed input $J(t)$ will impose the θ and τ time scales and also the number of temporal neurons that will be emulated. When the single node receives the preprocessed input $J(t)$, it receives sequentially the different incoming signals of the neurons it should emulate. Its response $x(t)$ is thus a sequence of the temporal neuron responses.
The time trace $x(t)$ is the equivalent of a neural network response. To satisfy to reservoir computing this time trace should have separability, approximation and fading memory. These properties will be given by the physics of the single node. The next subsection presents the different properties that the single node should have to perform reservoir computing.

5.2.3 *Key properties of the single node*

In order to ensure approximation property, the node should not have chaotic behavior and should not be too noisy.
The separation property in a spatial reservoir depends on the non linearity of neurons, the interconnection between neurons and the number of neurons. The number of neurons is not constrained in general for the node (with a fixed θ, τ just becomes bigger when the number n of temporal neurons increases). In order to ensure a non linear projection, the node needs to have itself a non-linear behavior. And finally to have interconnection between the temporal neurons, the different temporal states of the node should be linked together. In practice this is obtained by choosing a node with some memory mechanism which ensures that the present temporal state is linked to past temporal states. Two main memory mechanisms are classically leveraged to create interconnections between temporal neurons. The first one is to choose a node which has a relaxation in its response.

While the node is in a transient regime, its state still depends on the initial conditions. So the present state still depends on the past state. If θ is chosen smaller than the relaxation time, at time t the temporal state still depends on the response at time $t - \theta$ (so x_i depends on x_{i-1}). So in that case the preprocessed input $J(t)$ should be designed to suit the properties of the single node. Another common way to add memory is to use a feedback loop. The response of the node is stored in the delay line and reinjected later, thus creating a feedback.
These two mechanisms are also used for the fading memory. So in conclusion, the main ingredients for the node are the low noise, the non linearity and the memory. It is important to note that the geometry of the connections between the temporal states has always a ring shape (because past inputs influence present inputs).

5.2.4 *Equivalence between time and space*

With two time scales θ and τ the input driven non-linear node can emulate the behavior of a neural network but with a particular topology (ring shape). The equivalence between spatial and temporal networks is given by the following equation:

$$\mathbf{x}(k) = \begin{pmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_i(k) \\ \vdots \\ x_n(k) \end{pmatrix} = \begin{pmatrix} x((k-1)\tau + \theta) \\ x((k-1)\tau + 2\theta) \\ \vdots \\ x((k-1)\tau + i\theta) \\ \vdots \\ x(k\tau) \end{pmatrix} \tag{62}$$

After measuring the dynamical response $x(t)$ of the single non-linear node, one can retrieve the equivalent response of the reservoir by sampling values of the time trace $x(t)$ every θ time, according to equation 62. The rest of the process (training and inference) is then the same as with a classical spatial reservoir seen in previous section.

5.3 PREVIOUS HARDWARE IMPLEMENTATIONS OF SINGLE NODE RESERVOIR COMPUTING

The main requirement for reservoir computing is the non-linearity of nodes. Moreover, the reservoir computing with a single non-linear node simplifies drastically the experimental implementation. In this context, many experimental demonstrations were performed [239, 244, 245, 246, 247, 248, 249, 250, 251, 252]. Most of these demonstrations were made in optics and photonic, using a single non-linear node and a delayed feedback loop [244, 245, 246, 247, 253, 254, 250, 251, 252]. Some other works demonstrated reservoir computing with other systems, such as bucket of water, soft-robotic arm [249] or cat cortical brain [255], or VLSI [256], but the tasks solved were rather simple. Most of the photonic implementations use a phase modulator as single non linear node [244, 245, 246, 251, 247, 252, 254, 250]. In this system

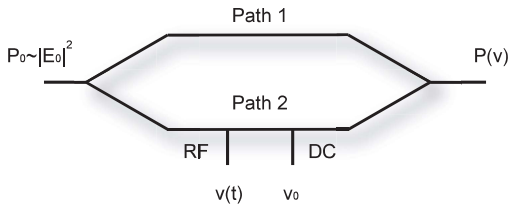

Figure 39: Scheme of Mach-Zehnder modulator. One arm of the modulator is modulated by 2 electrodes, an RF electrode and a DC electrode. The former contains the input and the feedback signal, the latter sets the offset phase of the nonlinearity. Extracted from Appeltant et al [257].

(Figure 39), the output power depends non linearly on the input power and the voltage imposed on the second arm of the modulator. The output power can be rewritten as:

$$P(v(t)) = P_0 sin^2(\pi v(t)/(2V_\pi) + \phi) \tag{63}$$

where P_0 is the input power of the laser, $v(t)$ is the control voltage of the modulator, V_π is the efficiency of the modulator and ϕ is an arbitrary phase. In practice, the preprocessed input $J(t)$ is converted into an oscillation voltage $V(t)$, and the Mach-Zehnder modulator receives a control voltage $v(t) = V(t) + V_0$, where V_0 is a bias current. The light power P, which is read using for instance a photodiode, is

then a non-linear transformation of the preprocessed input $J(t)$. The memory mechanism comes from delay lines, which inject the signal into various parts of the circuit: in the non-linear node [244, 246], after it [245, 251, 247], in an other non linear node [254, 250], or in a passive cavity [252]. Figure 40 is a picture of the implementation used in [246].

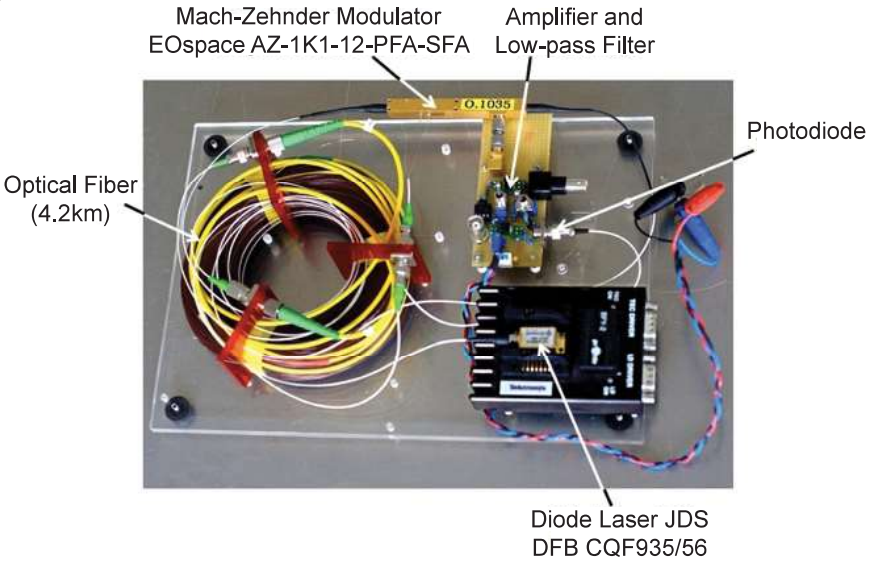

Figure 40: Picture of the opto-electronic reservoir computing set-up used in [246]. Extracted from [257].

The key elements here are the non linear Mach-Zehnder modulator, the delay loop made of the optical fiber, the amplifier and the low-pass filter, and the read-out photodiode. The system is powered by a diode laser. These experimental implementations could perform successfully complex tasks such as time series prediction [245, 251, 247, 252, 254] or speech recognition [244, 246, 251, 247, 252]. The strength of optics is the speed of computation [251, 250] and delay information with low loss in optical fibers. Moreover, this implementation uses only standard photonic components from telecommunications which are off-the-shelf and already well optimized. But such devices are hardly scalable because components such as phase modulators are centimeters big, kilometers of optical fibers are used and the use of lasers is quite energy intensive (Figure 40). A photonic spatial reservoir has been proposed [248], but the connection length on this 16 nodes chip was 2 cm, and it is predicted be downscaled only to 200 μm, which is still

too large for a fully analog brain-inspired chip. Finally, theoretical works propose to use skyrmions [258] and quantum systems [259] for reservoir computing.

5.4 CONCLUSION

In conclusion of this chapter, the principle of reservoir computing is to leverage the response of a recurrent neural network to map the initial problem in a higher dimension state where it is more likely to become linearly separable. To achieve this task, the reservoir should have separability, approximation and fading memory properties. Then learning to solve the problem is a linear regression problem.
This neural network architecture can be implemented using the dynamics of a single non-linear node which has some memory mechanism. It also requires preprocessing the input signal.
Complex tasks were performed experimentally using a single non-linear node with different physical systems (notably optics and photonics). But these systems are difficult to scale down and generally their power consumption may be relatively high. In the following chapters we will perform reservoir computing with a single spin-torque nano-oscillator as non-linear node.

6

SPINTRONIC IMPLEMENTATION OF RESERVOIR COMPUTING

6.1 EXPERIMENTAL IMPLEMENTATION

6.1.1 *Samples*

The samples used for this demonstration are fabricated by our collaborators from National Advanced Institute for Science and Technology. As mentionned earlier in chapter 4, they are magnetic tunnel junctions (MTJ) with a vortex magnetization in the free layer. These samples were selected for their excellent signal over noise ratio properties [173]. A schematic of the sample is shown in Figure 41. The

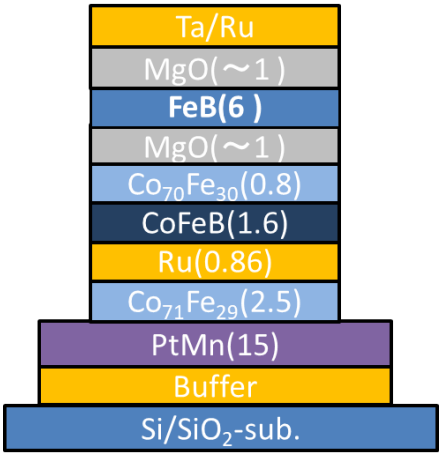

Figure 41: Schematic of the sample stack (thickness are indicated in nanometers).

pinned layer is made of a synthetic artificial antiferromagnet (SAF) in order to obtain a large spin polarization (see chapter 4). The SAF is $PtMn(15)/Co_{71}Fe_{29}/Ru/Co_{71}Fe_{20}B_{20}/Co_{70}Fe_{30}$ stack, where $PtMn$ is an antiferromagnet, $Co_{71}Fe_{29}$ is the first ferromagnet pinned through exchange bias interaction to the antiferromagnet. The sec-

ond ferromagnet $Co_{71}Fe_{20}B_{20}/Co_{70}Fe_{30}$ interacts with the first one through RKKY interaction. The Ru layer thickness is designed so the RKKY interaction forces the two ferromagnet to have magnetizations in opposite direction. The MgO tunnel barrier and FeB free layer were selected for high TMR ratio (135%) and low α ($\alpha_{FeB} \approx 0.07$). The magnetic tunnel junction (MTJ) films have a stacking structure of $buffer/PtMn(15)/Co_{71}Fe_{29}(2.5)/Ru(0.9)/Co_{71}Fe_{20}B_{20}(1.6)/Co_{70}Fe_{30}(0.8)/MgO(1)/Fe_{80}B_{20}(6)/MgO(1)/Ta(8)/Ru(7)$ (with thicknesses given in parenthesis in nanometres) and were prepared by ultrahigh vacuum (UHV) magnetron sputtering. After annealing at 360 C for 1 h, the resistance-area products (RA) were approximately 3.6 $\Omega/\mu m^2$. Circular-shape MTJs with a diameter of approximately 375 nm were patterned using Ar ion etching and e-beam lithography. The resistance of the samples is close to 40 Ω and the magneto-resistance ratio is about 135% at room temperature. For the dimensions used here (length L=6 nm and radius R=188 nm), the FeB layer has a remanent vortex magnetization. Under d.c. current injection, the core of the vortex steadily gyrates around the center of the dot with a frequency in the range 250-400 MHz for the oscillators we consider here. Vortex dynamics driven by spin torque are well understood (chapter 4), well controlled and have been shown to be particularly stable ([181]).

6.1.2 *Measurement set-up*

Measurement set-up principle is shown in Figure 42. The experimental pre-processed input signal V_{in} is generated by a high-frequency arbitrary-waveform generator and injected as a current through the magnetic nano-oscillator. The sampling rate of the source is set to 200 MHz (20 points per interval of time θ) for the spoken-digit recognition task and 500 MHz (50 points per interval of time θ) for the classification of sines and squares. The peak-to-peak voltage variation in the input signal is 500 mV, which corresponds to peak-to-peak current variations of 6 mA (part of the incoming signal is reflected owing to impedance mismatch between the sample and the circuit). The bias conditions of the oscillator are set by a d.c. current source and an electromagnet that applies a field perpendicular to the plane of the magnetic layers. These bias conditions impose the operating point of

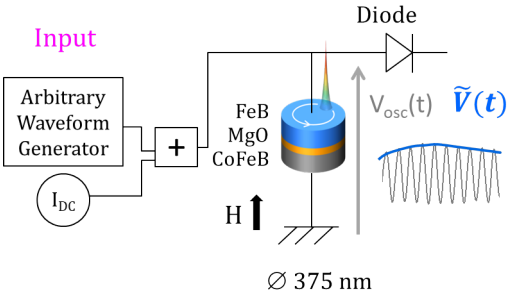

Figure 42: Principe of the measurement set-up: A magnetic field delivered by an electromagnet (not represented) and DC source fix the operating point of the oscillator. The signal to analyze is sent by an arbitrary waveform generator. The spin-torque emits an oscillating voltage. A diode allows extracting the amplitude of the oscillations.

the oscillator. The oscillating voltage emitted by the nano-oscillator is rectified by a planar tunnel microwave diode, with a bandwidth of 0.1-12.4 GHz and a response time of 5 ns. The input dynamic range of the diode is between 1 μW and 3.15 mW, corresponding to a d.c. output level of 0-400 mV. We use an amplifier to adjust the emitted power of the nano-oscillator to the working range of the diode. The output signal is then recorded by a real-time oscilloscope.

In practice, the spin-torque oscillator is connected in reflection configuration. The actual circuit (see Figure 43) is slightly more complicated than the previous schematic. Firstly, in order to decouple the d.c source and the incoming alternating signal from the arbitrary waveform generator, the oscillator is connected to a bias tee. The same capacity arm is used to send the input signal and receive the oscillator response. The arbitrary waveform generator is thus connected through a power splitter. At the power splitter interface, only half of the input signal is sent to the oscillator. The remaining other half of the input signal is then suppressed by a subtraction process: the second channel of the arbitrary waveform generator send the opposite of the undesired signal. The second channel of the arbitrary waveform generator is also connected to the rest of the circuit through a power divider. Subtraction is achieved by ensuring that these two signals are in phase (delay is controlled by tuning the length of the cables).

After subtraction, a circulator is placed. Its goal is to suppress the residual signal due to an imperfect subtraction (due to a small phase difference between the two signals). In practice, the circulator acts like a band-pass filter, which suppresses the lower frequency noise coming from a bad substraction. Finally after the circulator, only the signal of the oscillator is measured (noise induced by a bad subtraction represents about only 1% of the final signal). Then as described above, the signal is preamplified before passing through a diode which rectifies it. At the end the oscilloscope measures the amplitude of the oscillations which is the variable used for the computation.

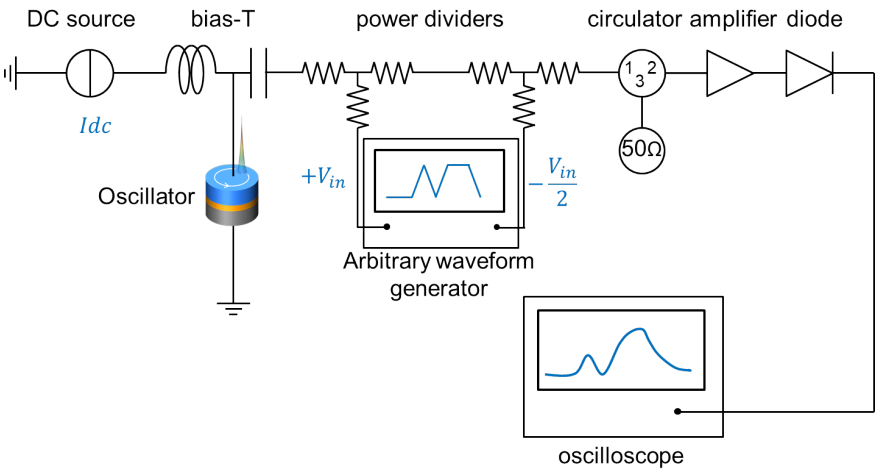

Figure 43: Detail of the measurement circuit: The oscillator is connected in reflection mode. The d.c. source signal and arbitrary waveform signal are decoupled by the bias-T. Injection of the arbitrary waveform signal is achieved with a power divider. A second power divider allows subtracting residual signal of the arbitrary waveform generator. A circulator is used as a band-pass filter. The oscillator signal is pre-amplified before going through a tunnel diode. An oscilloscope allows measuring the dynamical signal.

6.2 PHYSICAL PROPERTIES OF THE OSCILLATOR USED FOR COMPUTATION

The two main properties of the spin-torque nano-oscillator used for computation are the non linearity of the oscillation amplitude with

the input dc current and the relaxation of the oscillation amplitude. These two properties are necessary to ensure the separation property and the fading memory when reservoir computing is performed.

6.2.1 *Non-linearity of the oscillations amplitude*

As it has been seen previously in chapter 5, a reservoir realizes a non linear transformation of an input signal. This non-linearity allows the reservoir to project the initial problem in a higher dimensional state where different classes can be linearly separated (separation property). The variable used for computation is the amplitude level

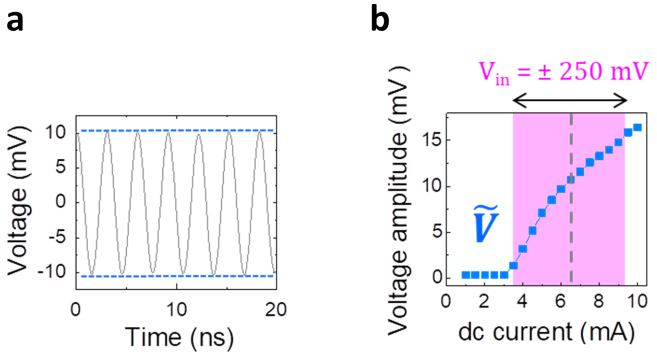

Figure 44: a, Measured a.c. voltage emitted by the oscillator as a function of time, $V_{osc}(t) = \widetilde{V}(t)cos(\omega t + \phi)$, for a steady current injection of 7 mA at an external magnetic field $\mu_0 H = 430$ mT. The dotted blue lines highlight the amplitude $\widetilde{V}$. b, Voltage amplitude $\widetilde{V}$ as a function of d.c. current I at $\mu_0 H = 430$ mT (blue squares). The purple shaded area highlights the typical excursion in the voltage amplitude that results when an input signal of $V_{in} = \pm 250$ mV is injected (here for $I = 6.5$ mA (vertical dotted line) and $\mu_0 H = 430$ mT).

(Figure 44a). The non-linearity used in this demonstration is thus the non-linearity of the amplitude level depending on the d.c. current. Figure 44b shows this non-linearity for a magnetic field of 430 mT. Below a threshold current (which is around 3 mA for this sample at this magnetic field), the oscillator does not emit any signal because the spin torque is not sufficient to compensate the pinning and thus the spin polarized current does not move the vortex core. After this threshold

current, the vortex core gyrates. The amplitude of the oscillation is proportional to the radius of the vortex orbit s. The non-linearity of the amplitude evolves approximatively like $\lambda(I, H_\perp)\sqrt{I - I_{th}}$ (see chapter 4 section 4.2.2).
As mentioned in the previous section, the input signal delivered by the arbitrary waveform generator has an amplitude of 500 mV peak to peak which corresponds to a 6 mA variation of injected current. Thus the input will induce variations on a limited part of the non linear amplitude curve. This area is represented in magenta in the figure for a d.c. current of 6 mA. The observation of Figure 44 illustrates the effect of the bias current on oscillation amplitude response $\widetilde{V}$ to an a.c. current. By changing the d.c. current, one would move the pink area (which is centered on the bias d.c current value) and thus the typical excursion in the voltage amplitude which is explored.
Influence of the magnetic field is also crucial because it changes the shape of the non linear function itself. It changes the threshold current (for smaller fields the threshold current is higher) and the amplitude of the non linearity. Choice for an optimal d.c. current and an optimal magnetic field will be discussed in chapter 8.

6.2.2 *Relaxation of the oscillation amplitude*

The second essential property for single node reservoir computing is a form of memory. This memory is important both for ensuring the connectivity between temporal neurons and to ensure a fading memory. In this demonstration only the intrinsic memory of the oscillator which is due to the relaxation of the amplitude is used. When the vortex core quits its orbit, it does it progressively. Thus the orbit radius and the proportional amplitude of the oscillation also vary progressively. Figure 45 shows the variation of the oscillation amplitude when the oscillator is subjected to a varying input signal. The response of the oscillator is plotted in grey and the amplitude of the oscillation is highlighted in blue. Transitions in the input signal are much more abrupt than in the oscillation amplitude signal. The characteristic time of these changes is the relaxation time of the oscillator which is coarsely defined by the frequency and the damping factor ($T_{relax} \tilde{} \frac{1}{\alpha f}$ see chapter 5 section 4.2.1). For our sample, the relaxation time is around 200 ns, besides when the current is close to the threshold.

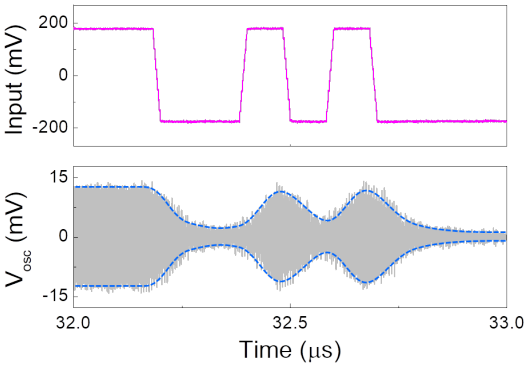

Figure 45: Relaxation of the oscillation amplitude: a. Input signal sent by the arbitrary waveform generator (magenta). b. Transient response of the oscillator. The emitted oscillating voltage is plotted in grey. The amplitude of the oscillation is figurated by dashed blue lines.

For the later regime, the relaxation time is larger but the emission amplitude is low and very noisy. So it is difficult to exploit this regime for computation. A detailed evaluation of the relaxation time will be made in chapter 10 section 10.1.
The relaxation is important to determine which time θ is used to emulate the response of a temporal neuron. In order to have connections between the temporal neurons, we should choose $\theta < T_{relax}$. In the results in chapter 7, $\theta = 100$ ns will be chosen. Discussion about this choice will be made in chapter 8.

7 RESULTS ON CLASSIFICATION TASKS

In this chapter we present the classification results obtained using a single vortex spin-torque nano-oscillator in the framework of single node reservoir computing [2]. Two different tasks were studied. First, sine versus square classification task was studied. The second task we studied is spoken digit recognition.

7.1 RESULTS ON SINE/SQUARE RECOGNITION TASK

7.1.1 *Task*

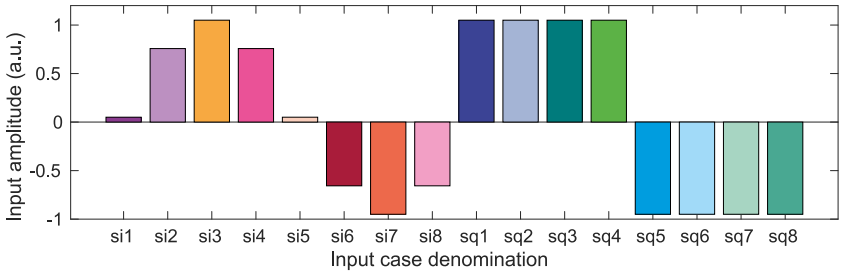

Figure 46: Pattern to recognize: the sine and square periods are discretized in 8 values called si1 to si8 (represented in warm colors) and sq1 to sq8 (represented in cool colors).

The sine and square waveform classification task has been already used in other studies [244] to evaluate the performance of reservoir computing based on a delayed feedback spin torque oscillator. The goal is to classify each point of the input as part of a sine by returning an output 0, or as part of a square by returning an output 1. Each period of sine and square is discretized in 8 points and therefore we have 16 cases to process (see Figure 46). We refer to the 8 points of the sine as si1-8 represented in warm colors and the 8 points of the square as sq1-8 represented in cold colors. Because of the symmetries the inputs take 5 different values in a sine period and two different values

in a square input. To return the same output value for 5 different input values of a sine (or 2 in the case of the square), the reservoir should have a non-linear behavior. In addition, in the sine period the 3rd and the 7th point (referred as si3 and si7 in Figure 46) have a value +1 and -1 that corresponds to the values taken by the input in the square. So in the absence of memory, when the input value is +1 or -1 it is impossible to know whether these points belong to a sine or to a square. Therefore this temporal pattern recognition task is not trivial because it needs both the non-linearity and the memory of a neural network.

7.1.2 *Protocol*

The input $u(k)$ is composed of 1280 points (160 randomly arranged periods of sine or square). The first half of the points are used for training (to find optimum output weights W) and the second half for testing. Different steps of the protocol are represented in Figures 47a-c. Figure 47 shows how the oscillator is driven by the input signal. The input u(k) (Figure 47a) is a sequence of discretized sine and square periods. This discrete input is then preprocessed. The preprocessed input J(t) for a sine and square period is represented in Figure 47b. An input with only 12 temporal neurons is represented here in order to clarify the figure. Here the mask contains only binary values +1 and -1 and a reservoir with 12 temporal neurons is emulated. In the next subsection results for a 24 temporal neurons reservoir will be presented. The time step θ is taken here to be equal to 100 ns which was found to be optimal. These points will be developed in chapter 8. When the oscillator receives this preprocessed input, it emits a transient response. The amplitude of the emitted oscillation measured experimentally is plotted in Figure 47c.

Figure 48a shows how to retrieve the higher dimensional mapping operated by the single oscillator driven by the preprocessed input. This step is done off line on a computer. Discrete points are sampled every time step θ. The values of these points correspond to the response of the temporal neurons. Once these value are selected, one obtains the reservoir state. To sample properly the temporal traces of the oscillation amplitude, they should be aligned with the preprocessed input (misalignment can result in bad classification).The rest of the

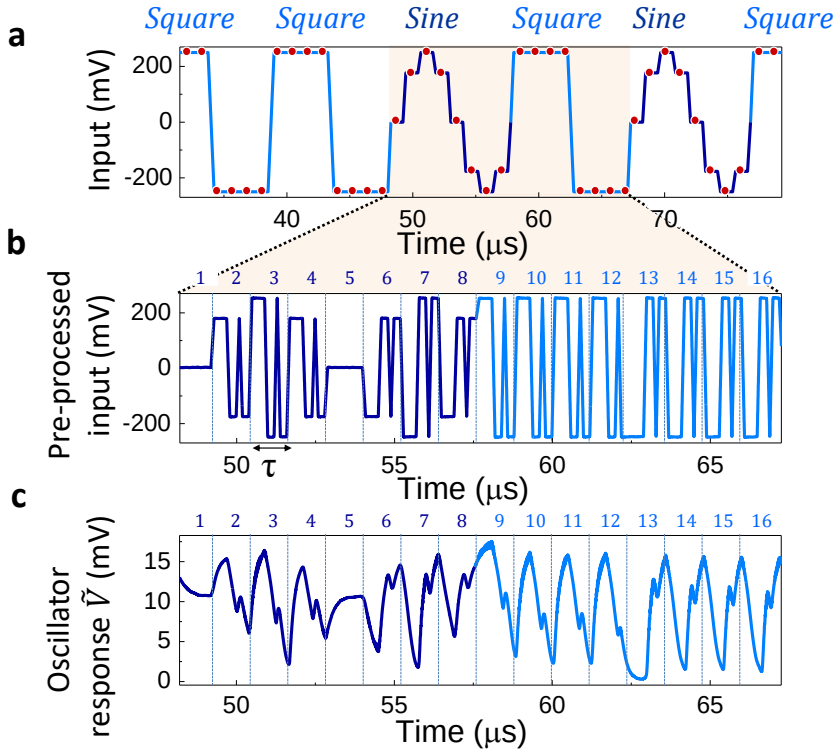

Figure 47: Sine square classification process: a. signal to classify composed of periods of sine and squares. The points to identify as belonging to sine or square are figurated in red. b. Preprocessed input signal for a sine (dark blue) and square (light blue) period. In this case a mask with 12 values is used. c. Oscillation amplitude transient response to the oscillator signal.

process is a standard reservoir computing procedure. The neuron responses are linearly combined to reconstruct the output. This step is shown in Figure 48b. The coefficients of the linear combination are obtained with the standard learning procedure seen in chapter 5. At the end the single oscillator with time multiplexing emulates the response of a ring shape recurrent neural network. This architecture is represented in Figure 48c. Response of this equivalent neural network is used to classify the sine and square inputs.

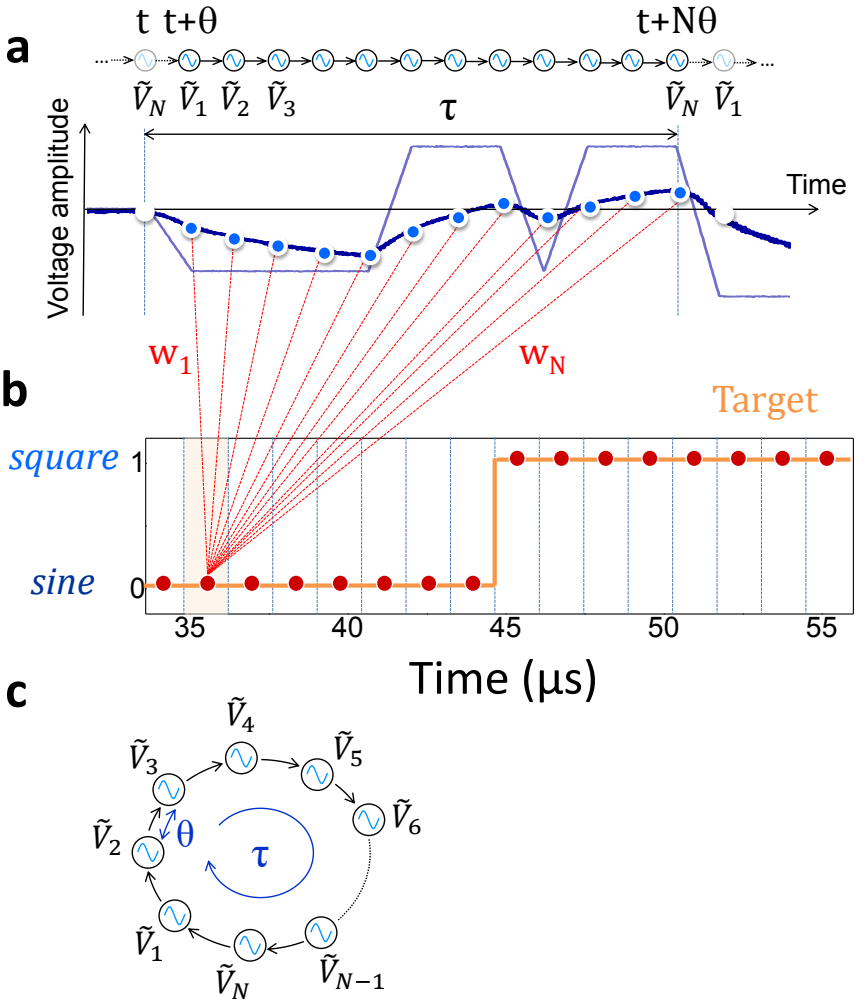

Figure 48: Higher dimension mapping and reconstruction of the output: a. Oscillator voltage amplitude $\widetilde{V}$ changes corresponding to a single time segment τ: Here, 12 neurons (12 samples $\widetilde{V}_i$ separated by the time step θ) are used to construct the output. b. Target for the output reconstructed from the voltages $\widetilde{V}_i$ in each time segment τ: output $\sum_{i=1}^{N} w_i \widetilde{V}_i$. c. The transient states of the oscillator give rise to a chain reaction emulating the neural network with a ring structure.

7.1.3 *Set point dependent results*

Figure 49 shows the reconstructed output obtained by experimentally emulating a 24-neuron network. The root mean square (rms)

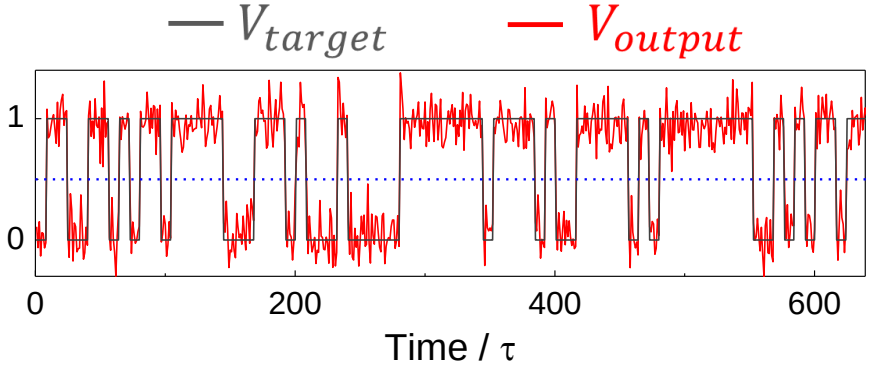

Figure 49: Reconstructed output (red) and target (grey) in response to an input waveform with 80 randomly arranged sines and squares. The magnetic field is 447 mT, and the applied current 7.2 mA. The results are based on 24 neurons separated by $\theta = 100$ ns.

deviation between target and output is 11 %, which is small enough to distinguish between sines and squares without any error (perfect classification) for the chosen choice of parameters: d.c. current = 7.2 mA, magnetic field H= 447 mT, input amplitude V_{in}= 500 mV (equivalent to 6 mA peak to peak). Indeed as it can be observed in Figure 49, if we trace a threshold line (in blue) at 0.5, all the outputs for square inputs are over this threshold and all the points for sine inputs are under this threshold. This perfect classification of sine and squares is obtained for an optimal operating point (imposed by the values of the bias d.c. current and the bias magnetic field). As it can be seen in Figure 50, the quality of pattern recognition, characterized by the root-mean-square of deviations between the reconstructed output and the target, varies from 10% to more than 30% depending on the bias conditions. The interpretation of the optimal operating point conditions will be made in chapter 8. After selecting this region of magnetic field and d.c. current leading to high performance of the oscillator for sine/square classification, we moved to the more complex task of spoken digit recognition.

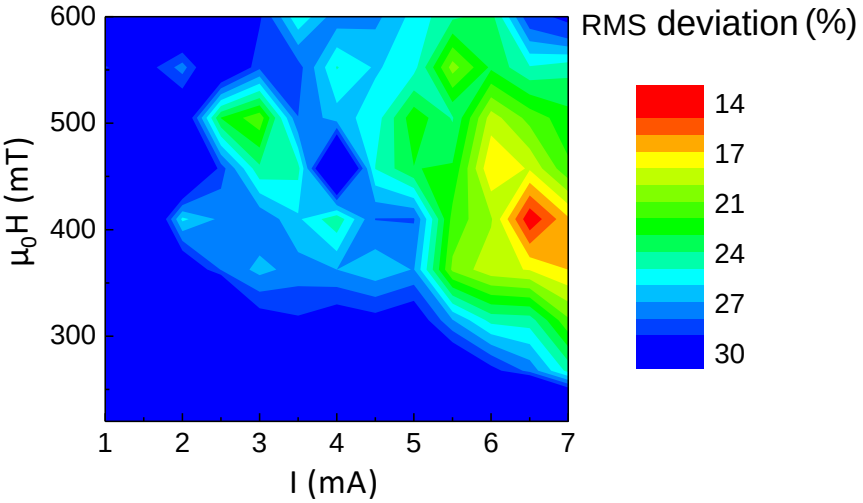

Figure 50: Rootmean-square (r.m.s.) deviation of output-to-target deviations: map as a function of d.c. current I and magnetic field $\mu_0 H$. For these results, V_{in}= 500 mV and θ= 100 ns.

7.2 RESULTS ON SPOKEN DIGIT RECOGNITION

7.2.1 *Task*

Spoken digit recognition is a widely used benchmark task in the hardware reservoir computing community (see section 5.3). The goal of the task is to recognize digits from audio waveforms produced by different speakers. For this task, the inputs are taken from the NIST TI-46 data corpus. The input consists of isolated english spoken digits said by five different female speakers. Each speaker pronounces each digit ten times. The 500 audio waveforms are sampled at a rate of 12.5 kHz and have variable time lengths. Only female speakers are chosen to benchmark with the literature. Adding male and children would make the task more difficult to solve because it would add more different tones of voices and thus increase the dispersion of the data to classify.

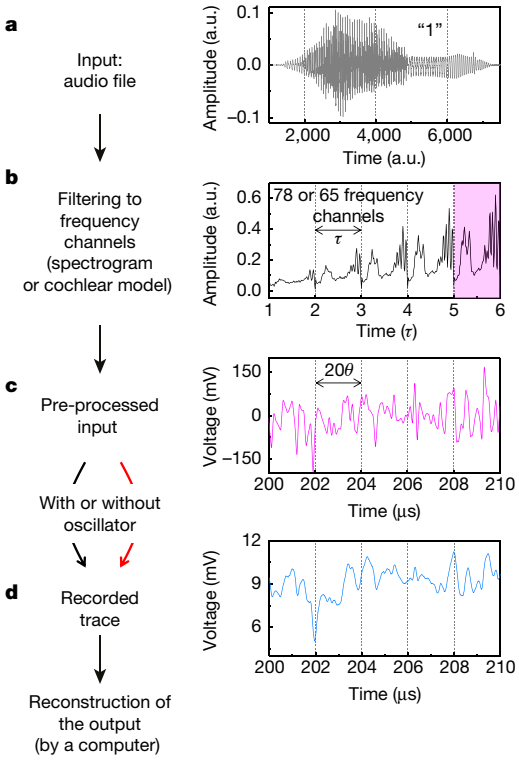

Figure 51: Protocol for spoken digit recognition. a-d, Principle of the experiment. a, Audio waveform corresponding to the digit 1 pronounced by speaker 1. b, Filtering to the frequency channels for acoustic feature extraction. The audio waveform is divided in intervals of duration τ. The cochlear model filters each interval into 78 frequency channels (65 for the spectrogram model), which are then concatenated as 78 (65) values for each interval, to form the filtered input. c, Pre-processed input (transformed from the purple shaded region in b). The filtered input is multiplied by a randomly filled binary matrix (masking process), resulting in 400 points separated by a time step θ of 100 ns in each interval of duration τ ($\tau = 400\theta$). d, Oscillator output. The envelope $\widetilde{V}(t)$ of the emitted voltage amplitude of the experimental oscillator is shown ($H = 430$ mT, $I = 6$ mA). The 400 values of $\widetilde{V}(t)$ per interval τ ($\widetilde{V}_i$, sampled with a time step θ) emulate 400 neurons. The reconstructed output '1', corresponding to this digit, is obtained by linearly combining the 400 values of $\widetilde{V}_i$, sampled from each interval τ.

7.2.2 *Protocol*

The recognition of audio waveforms requires a time-domain to time-frequency domain transformation, prior the reservoir. For this purpose we used two different filtering methods: spectrogram and cochlear models. Both filters break the word into several time intervals N_τ of duration τ and analyze the frequency content in each interval τ through either a Fourier transform (spectrogram model; 65 channels, $N_\tau \in 24, ..., 67$) or a more complicated nonlinear approach (cochlear model; 78 channels, $N_\tau \in 14, ..., 41$). A detailed comparison of these two methods will be done in chapter 9. The input for each word is composed of an amplitude for each of the $N_f = 65$ or $N_f = 78$ frequency channels times N_τ time intervals. This input is pre-processed by multiplying the frequency content for each time interval by a mask matrix containing $N_f \times N_\theta$ random binary values, giving a total of $N_\tau \times N_\theta$ values as input to the oscillator (Figure 51). Here, we are modeling $N_\theta = 400$ input neurons, each of which is connected to all of the frequency channels for each time interval. Each preprocessed input value is consecutively applied to the oscillator as a constant current for a time interval of $\theta \approx 100$ ns. This time is short enough to guarantee that the oscillator is maintained in its transient regime so the emulated neurons are connected to each other, but is long enough to let the oscillator respond to the input excitation. The amplitude of the a.c. voltage across the oscillator is recorded for offline post-processing (Figure 51d). As presented in chapter 5, the post-processing of the output consists of two distinct steps (training and testing). The goal of training is to determine a set of weights $w_{i,\theta}$, where i indexes the desired digit. These weights are used to multiply the output voltages $\widetilde{V}_\theta$ to give $10N_\tau$ output values, which are then averaged over the N_τ time intervals to give 10 output values y_i, which should ideally be equal to the target values $y_i = 1.0$ for the appropriate digit and 0.0 for the rest. In the training process, a fraction of the utterances are used to train these weights; the rest of the utterances are used in the classification process to test the results. The optimum weights are found by minimizing the difference between $\widetilde{y}_i$ and y_i for all of the words used in the training. In practice, optimal values are determined by using techniques for extracting meaningful eigenvalues from singular matrices such as the linear Moore-Penrose

pseudo-inverse operator (denoted by a dagger symbol † as seen in chapter 5). If we consider the target matrix $\widetilde{Y}$, which contains the targets $\widetilde{y_i}$ for all of the time steps τ used for the training, and the response matrix S, which contains all neuron responses for all of the time steps τ used for the training, then the matrix W, which contains the optimal weights, is given by $W = \widetilde{Y}^{\dagger}$. This step is performed on a computer and takes several seconds. During the classification phase, the ten reconstructed outputs corresponding to one digit are averaged over all of the time steps τ of the signal, and the digit is identified by taking the maximum value of the ten averaged reconstructed outputs. The averaged reconstructed output that corresponds to the considered digit should be close to 1 and the others should be close to 0. The efficiency of the recognition is evaluated by the word success rate, which is the rate of digits that are correctly identified. The training can be done using more or fewer data (here "utterances"). We always trained the system using the ten digits spoken by the five speakers. The only parameter that we changed is the number of utterances used for the training. If we use N utterances for training, then we use the remaining $10 - N$ utterances for testing. However, some utterances are very well pronounced whereas others are hardly distinguishable. As a consequence, the resulting recognition rate depends on which N utterances are picked for training in the set of ten (for example, if $N = 2$, then the utterances picked for training could be the first and second, but also the second and third, or the sixth and tenth, or any other of the $10!/(8!2!)$ combinations of 2 picked out of 10). To avoid this bias, the recognition rates that we present here are the average of the results over all possible combinations. The error bars correspond to the standard deviation of the word recognition rate. In order to see the contribution of the spin-torque oscillator in the recognition process we compare the results obtained from the oscillator time traces with a control trial. During the control trial, the pre-processed inputs are used for reconstructing the output on a computer directly, without going through the experimental set-up.

7.2.3 *Preprocessing dependant results*

The improvement shown in the experimental results over the control results (see Figure 52a) indicates that the spin-torque nano-oscillator

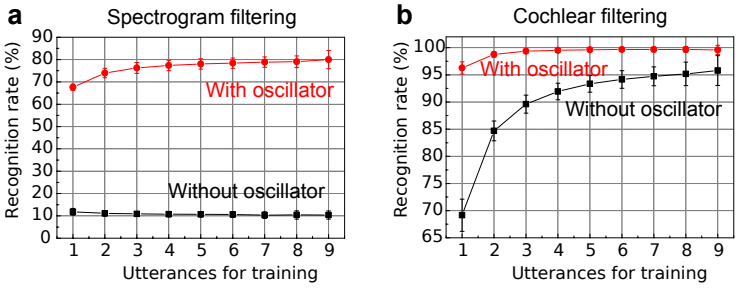

Figure 52: a,b, Spoken-digit recognition rates in the testing set as a function of the number of utterances N used for training for the spectrogram filtering (a; H= 430 mT, I= 6 mA) and for the cochlear filtering (b; H= 448 mT, I=7 mA). Because there are many ways to pick the N utterances, the recognition rate is an average over all $10!/[(10-N)!N!]$ combinations of N utterances out of the 10 in the dataset. The red curves are the experimental results using the magnetic oscillator. The black curves are control trials, in which the pre-processed inputs are used for reconstructing the output on a computer directly, without going through the experimental set-up. The error bars correspond to the standard deviation of the recognition rate, based on training with all possible combinations.

greatly improves the quality of spoken-digit recognition, despite the added noise that is concomitant to its nanometre-scale size. In this case, the extraction of acoustic features, achieved by Fourier transforming the audio waveform over finite time windows, plays a minimal part in classification. Without the oscillator (black line), the recognition rates are consistent with random choices; with the oscillator (red line), the recognition rate is improved by 70%, reaching values of up to 80%. This example highlights the crucial role of the oscillator in the recognition process. Using the cochlear filtering (Figure 52b), which is the standard in reservoir computing and has been optimized on the basis of the behavior of biological ears, we achieve recognition rates of up to 99.6%, as high as the state-of-the-art. Compared to the control trial, the oscillator reduces the error rate by a factor of up to 15. Our results with a spin-torque nano-oscillator are therefore comparable to the recognition rates obtained with more complicated electronic

or optical systems (between 95.7% and 99.8% for the same task with cochlear filtering).

7.3 CONCLUSION

In this chapter, two tasks were used to evaluate the performance of reservoir computing using the dynamic of a spin-torque nano-oscillator: sine/square classification and spoken digit recognition. Sine/square classification is a simpler task but allows to test the non-linear behavior and the memory of the reservoir which are the key features to achieve good classification on more complex tasks. Using 24 temporal neurons, a systematic study of classification for different magnetic fields and d.c. current bias conditions was led. In the best case a 10% root-mean-square deviation between the reconstructed output and the target was obtained, which allows perfect classification of sine and square inputs. Best bias conditions were selected to move on to the more complex task of spoken digit recognition. Spoken digit recognition requires frequency filtering of audio file prior to sending the input to the oscillator. Two filtering methods where studied: first a simple linear spectrogram method and a more complex cochlear decomposition to benchmark our result with the existing litterature. Using spectrogram method, the oscillator improves the recognition up to 70% and the overall success rate is 80% by training on 90% of the data. This result stresses the critical role of the nano-oscillator for recognition. Using a cochlear decomposition, 99.6% success rate was reached which is state of the art results for both numerical and hardware methods. Also it is important to note that training a linear classifier directly on a cochleogram allows to reach a success rate of 96%. Thus cochlear decomposition already separates a part of the input. The results presented in this chapter are the first experimental demonstration of neuromorphic computing performed with a nanoscale "neuron". For this demonstration, experimental parameters such as the filtering method, the temporal time scale θ and the operating point (d.c current and field) were specially selected to achieve good classification results. Next part will elucidate the influence of these different parameters and give guidelines to choose them appropriately.

Part IV

INFLUENCE OF THE EXPERIMENTAL PROTOCOL ON THE FINAL CLASSIFICATION RESULT

8

OPTIMIZING THE EXPERIMENTAL PARAMETERS AND DATA PROCESSING FOR IMPROVED CLASSIFICATION

In this chapter, the different parameters that can be tuned to optimize the recognition rate are presented. The performance are evaluated on the sine and square classification task which is easier to deal with. The first section presents the influence of the time step θ. The second section shows how to bring memory with a numerical shift. The third section analyzes the effect of the operating point.

8.1 THETA DEPENDENCE AND INPUT AMPLITUDE

As it was explained in chapter 7, the memory and connectivity in single oscillator reservoir computing is obtained by driving the oscillator with a fast varying preprocessed input. The rate of the input variation is θ. In this part, we discuss how to optimize θ to obtain the best classification performances. The results presented are obtained for the operating point $\mu_0 H = 440$ mT and $I = 7.0$ mA. Figure 53 illustrates the dynamical response of the oscillation envelope for different values of θ and different amplitudes of the input. The upper left figure (Figure 53a) shows the input signal sent to the oscillator by the arbitrary waveform generator (AWG). The amplitude of the signal is here normalized but three different amplitude values where tested (300 mV, 400 mV and 500 mV peak to peak). The time is also normalized and given as numbers of θ. Other figures represent the oscillation amplitude response for respectively $\theta = 25$ ns (Figure 53 b), $\theta = 100$ ns (Figure 53c) and $\theta = 300$ ns (Figure 53d). For $\theta = 25$ ns, the time rate of the input is much shorter than the intrinsic relaxation time of the oscillator measured around 200 ns ($\theta = T_{relax}/8$) so the oscillator stays in the transient regime all the time. The memory effect is in this case strong but the amplitude of the emitted oscillation amplitude signal is quite small. Indeed the oscillator does not have the time to reach higher values. Amplitude

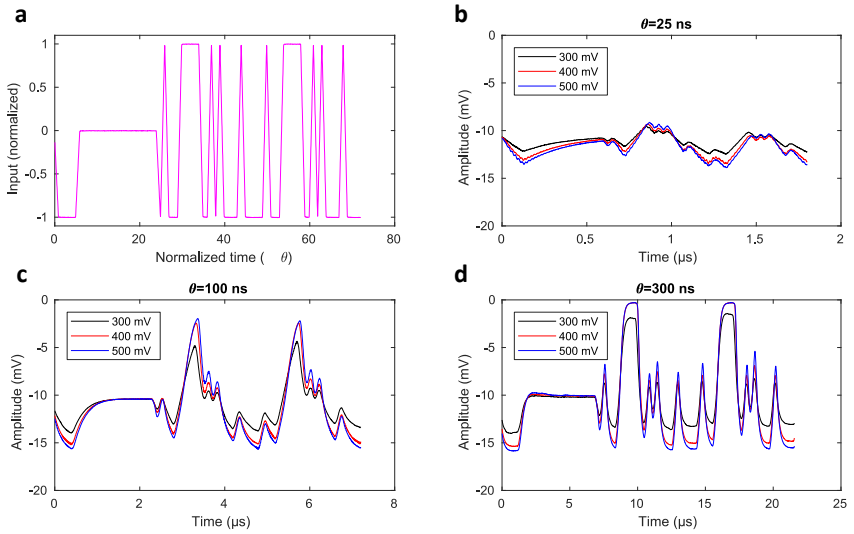

Figure 53: Effect of θ on the dynamics of oscillation amplitude: a, normalized input signal to the oscillator (magenta curve). In order to be general for all the following cases, the time scale is normalized by θ and the voltage is normalized by the amplitude of the input. b,c,d, $\widetilde{V}$ oscillation amplitude variation of the oscillator response for three level of input 300 mV (black), 400 mV (red) and 500 mV (blue) in the case of time step $\theta = 25$ ns (b), $\theta = 100$ ns (c) and $\theta = 300$ ns (d). Measurements are made at $\mu_0 H = 440$ mT and $I = 7.0$ mA.

of the oscillation also depends of the input peak to peak amplitude and for an input amplitude of 300 mV, the signal is particularly small. At the other extreme for $\theta = 300$ ns, superior to the oscillator relaxation time ($\theta = 1.5T_{relax}$), the measured amplitude response reaches its saturation regime and the oscillator loses the memory of what happened one θ in the past. But for such high θ, the amplitude of the signal is much larger because the oscillator reaches it saturation regime. Finally for an intermediate value $\theta = 100$ ns (approximatively $T_{relax}/2$), the oscillator still remains in a transient regime with a large amplitude. Intuitively, we expect that the trade off will be to choose θ allowing to have a decent amplitude (so a decent signal over noise ratio) and still have memory. It is important to note that a higher input amplitude for a fixed theta always improves the signal to noise ratio. Next Figure 54 presents the experimental RMS error result as a

function of θ. For 500 mV, increasing θ first decreases the RMS until

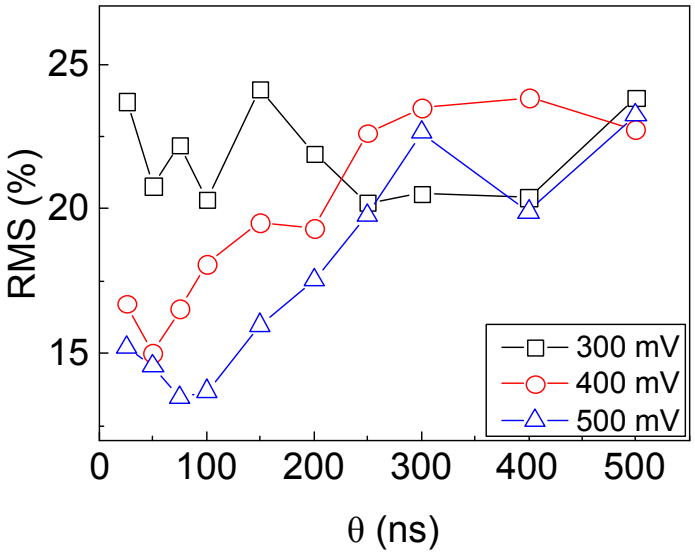

Figure 54: Classification performance depending of θ: Root mean square of output-to-target deviations as a function of the time step θ (separation between transient states of the oscillator $\widetilde{V}_i$) for different amplitudes of the input signal (300, 400 and 500) mV when the target is in exact phase with the input.

an optimum after which the RMS increase again. This optimum seems to be around 100 ns. In the first part the error decreases because the signal to noise ratio increases but after $T_{relax}/2$ the error increases again because of a loss of memory. So the causes of the error change depending on θ. This evolution trend is verified for an input amplitude of 400 mV. In that case, surprisingly the optimum seems to occur for a smaller θ of 50 ns. However the general result is always worse than for 500 mV input amplitude. Finally for an input amplitude of 300 mV, no clear trend is observed and the classification results are bad. Both a satisfying memory and a satisfying signal over noise ratio do not seem to be met for one of the θ conditions, for this particular voltage amplitude input.

8.2 IMPROVEMENT WITH A NUMERICAL SHIFT

This part explains how to overcome the memory problem, bringing artificially memory with a numerical shift. Since the output is reconstructed off-line after recording the whole response to inputs, it is possible to shift the target with respect to the input. In that case, some of the samples used for reconstructing the output belong to the previous segment. In other words, the current output is reconstructed partly from the present value of the input and partly from the last value of the input. This shift in the data selection is illustrated in Figure 55.

In Figure 55a we observe the classical selection of data which was already depicted in chapter 7. In this illustration we focus on the reconstruction of the second point of the output. Classically, the virtual neuron responses are selected in the time interval $[\tau, 2\tau]$ (orange area in Figure 55a) to reconstruct the second point of the output (orange area on the target). The virtual neuron responses are highlighted with blue circles and they are combined linearly to obtain the output value (red dashed line). The numerical shift is depicted in Figure 55b. In Figure 55b we choose a shift value of $\tau/2$, so the data selection area (orange area) is shifted of $\tau/2$ on the left compared with the target (the interval is here $[0.5\tau, 1.5\tau]$). Experimentally we observed the best results for this value of shift when the number of samples is evenly distributed between the past and current input value. In Figure 56, we evaluate the effect of a $\tau/2$ shift varying the θ value and the input amplitude.

This Figure 56 can be compared with previous section results in Figure 54 without numerical shift. At first glance it is noticeable that the shift has a positive effect for all the input amplitude values. We notice the change of scale for RMS compared with previous section. Here RMS is in the range 5-20%, instead of 10-25% in previous section. Much smaller RMS values are reached here. The shift improves the RMS result of approximatively 5% at low θ ($\theta < 100$ ns), where error is dominated by the noise. Even in these parts, some errors are due to memory since the shift improves the result. The effect of shift is even more striking after 100 ns. In contrary to the case without shift, the error does not increase again. For 400 mV and 500 mV input amplitude it stabilizes and for 300 mV it still decreases. For 400 mV

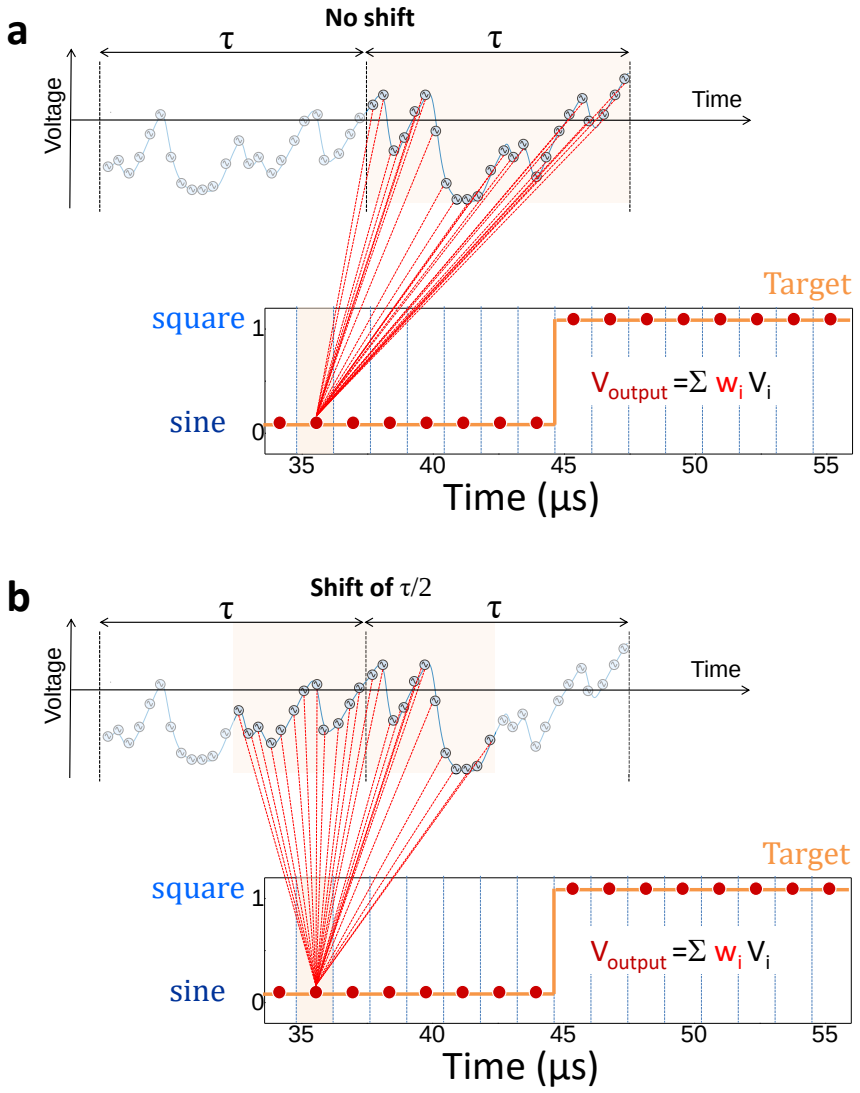

Figure 55: a. Classical selection of data for reconstruction of the output when the target is in exact phase with the input. The amplitude response between τ and $\tau + 1$ (orange area) is used to reconstruct the second output point (red dot in orange area) b. Selection of data in the case of a shift of $\tau/2$: the target is shifted by $\tau/2$ with respect to the phase. The amplitude response between $1/2\tau$ and $3/2\tau$ (orange area) is used to reconstruct the second output point (red dot in orange area).

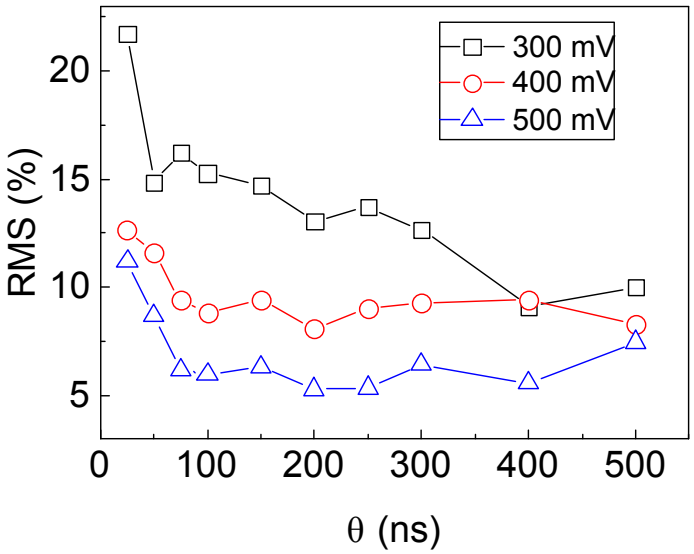

Figure 56: Classification performance depending on θ using a shift between the amplitude and the target of $\tau/2$: Root mean square of output-to-target deviations as a function of the time step θ (separation between transient states of the oscillator $\widetilde{V}_i$) for different amplitudes of the input signal (300, 400 and 500) mV when the target is shifted by $\tau/2$ with respect to the amplitude.

and 500 mV, the oscillation amplitude response already reaches with a θ of 100 ns a satisfying level to compute. For 300 mV input amplitude, the θ should be greater to obtain a satisfying signal over noise ratio. The fact that for all cases, the input does not increase again after 100 ns is in good agreement with the interpretation of the previous section. Without shift, after 100 ns, the error is dominated by a lack of memory. Adding a shift emancipate from memory problems so for all θ values, the error is dominated by the signal over noise ratio which increases and then stabilized increasing θ just like the error decreases and stabilized.

8.3 MAGNETIC FIELD AND D.C. CURRENT DEPENDENCE

8.3.1 *Illustration of different performances for different dynamical regimes*

In this section we illustrate the different dynamical responses that occur depending of the operating point choice.

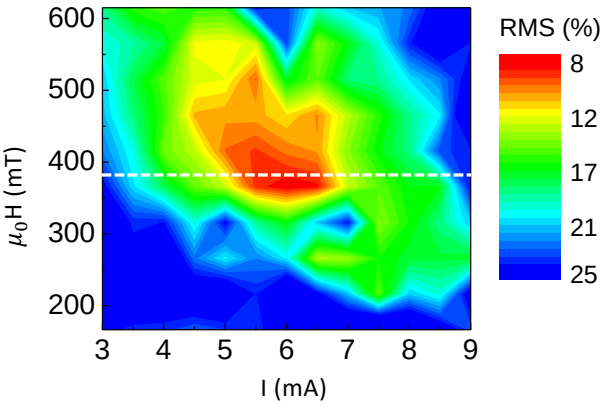

Figure 57: Root mean square of output-to-target deviations: map as a function of d.c. current I and magnetic field H. Here $V_{in} = 300$ mV and $\theta = 100$ ns is used. RMS map corresponds to the target shifted by $\tau/2$ with respect to the input. The dashed white line corresponds to a field of 380 mT.

Figure 57 shows the RMS performances for sine/square classification. Compared with Figure 50 in chapter 7, it is important to note two major differences. First is that input signal amplitude sent by the AWG is only 300 mV and not 500mV. This allows us to explore a larger range of d.c. current without risking to deteriorate the oscillator. Second point is that these results are given using the numerical shift. This numerical shift allows us to consider the performance independently of the oscillator memory since the memory is brought by the shift. The memory effect will be treated in next part (part v) of this manuscript. Compared with chapter 7, one can notice that the numerical shift allows better RMS performances. But we obtain qualitatively similar results with a best performance region between 300mT and 500mT and for a d.c. current between 5mA and 7mA. To illustrate the different dynamical regimes of the oscillator, we selected the magnetic

field 380 mT where best performances are obtained. The different oscillator amplitude transient responses to the input signal are plotted in Figure 58 at this field and for respective d.c. currents of 3.0 mA, 4.5 mA, 6.5 mA and 9.0 mA.

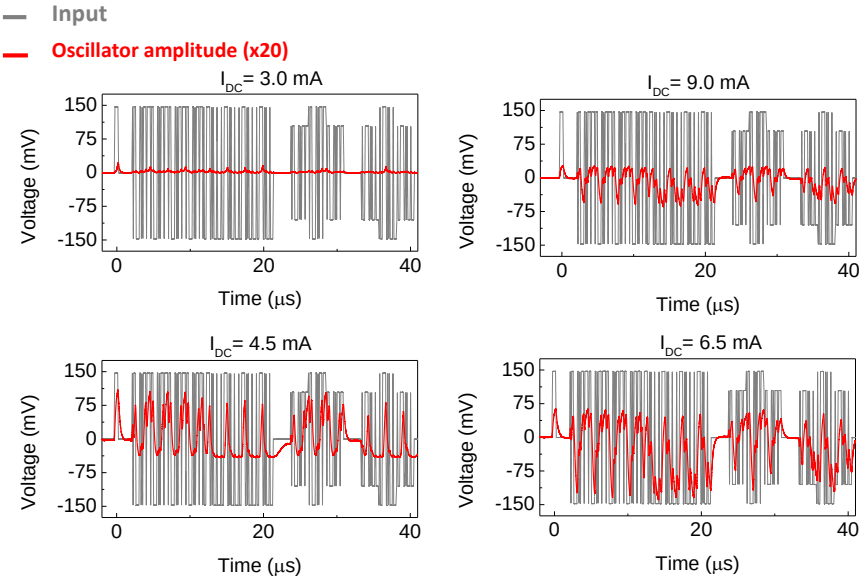

Figure 58: The oscillator voltage amplitude $\widetilde{V}$ curves (in red) in response to the input waveform (in gray) are plotted for selected d.c. currents (3, 4.5, 6.5 and 9) mA and magnetic field $H = 380$ mT. Here $V_{in} = 300$ mV and $\theta = 100$ ns is used.

These four values correspond to different RMS performances. 3.0 mA and 9.0 mA gives poor RMS results (over 20%). 4.5 mA give intermediate results around 15% RMS. And finally 6.5mA gives the best RMS results below 8%. Looking at the red time traces one can notice that all the d.c. current lead to different regimes with various amplitude (for example for 3 mA, the amplitude is very low, whereas for 6.5mA it is very high), various non-linearity, which can be seen in the assymetry of the response and various noise levels. Qualitatively, different operating points induce different regimes and RMS deviations. The oscillator performs well when it responds strongly to the time varying preprocessed input, with large amplitude variations in both positive and negative directions, V_{up} and V_{dw}, respectively (Figure 59a). On the other hand, it performs poorly when the noise in the oscillator

ΔV (the standard deviation of the noise in the voltage amplitude) is high (Figure 59b).

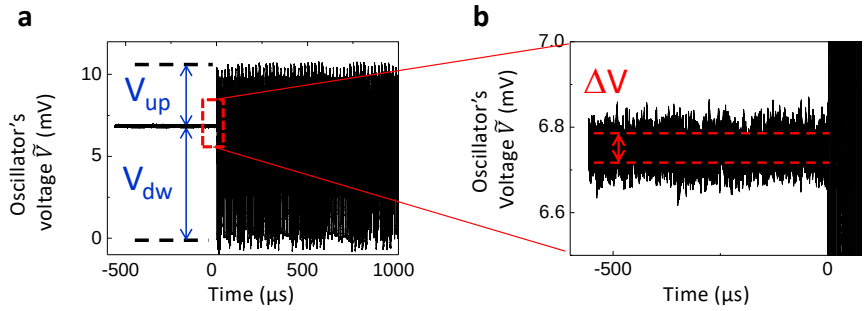

Figure 59: a. Amplitude variations in both positive and negative directions, V_{up} and V_{dw} b. Standard deviation of the noise in the voltage amplitude ΔV.

8.3.2 *Noise and amplitude ratio*

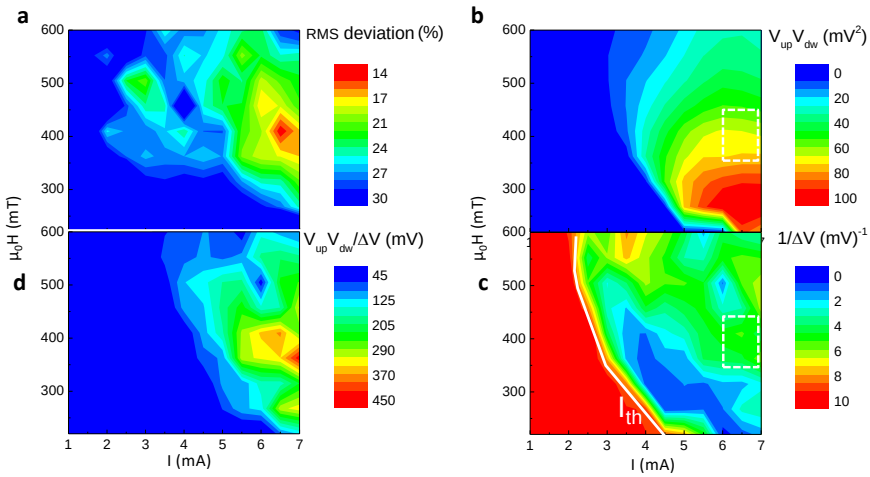

Figure 60: a. RMS deviation for V_{in}=500 mV as a function of the operating point. b. $V_{up}V_{dw}$ as a function of the operating point. c. $1/\Delta V$ as a function of the operating point. The white line correspond to the critical current I_{th} and the white rectangle corresponds to the region of smallest RMS. d. $V_{up}V_{dw}/\Delta V$

As shown in Figure 59, we extract these parameters from the time traces of the voltage emitted from the oscillator at each bias point,

and plot $V_{up}V_{dw}$ (Figure 60b) and $1/\Delta V$ (Figure 60d) as a function of the d.c. current I and field H. The red regions of large oscillation amplitudes in Figure 60b correspond to low magnetic fields, in which the magnetization is weakly confined, and to high currents, for which the spin torque on magnetization is maximal. The blue regions of high noise in Figure 60c correspond to areas just above the threshold current I_{th} for oscillation, in which the oscillation amplitude $\widetilde{V}$ is growing rapidly as a function of current and is becoming sensitive to external fluctuations. As can be seen by comparing Figure 60b and c, the range of bias conditions highlighted by the dotted white boxes (currents of 6-7 mA and magnetic fields of 350-450 mT) features wide variations in oscillation amplitudes and low noise. In this region, root-mean-square deviations below 15% are achieved, and there are no classification errors between sine and square waveforms. The similarity between the map of $V_{up}V_{dw}/\Delta V$ (Figure 60d) and that of the classification performance (Figure 60a) confirms that the best conditions for classification correspond to regions of optimal compromise between low noise and large amplitude variations. Importantly Figure 60a is the RMS obtained for V_{in}=500 mV and no shift. The necessity of a high signal-to-noise ratio for efficient neuromorphic computing, highlighted here for magnetic oscillators, is a general guideline that applies to any type of nanoscale oscillator. In the next section we explicit the origin of the different symmetries of the amplitude depending of the operating point.

8.3.3 *Amplitude level*

By modifying the operating point, one modifies the oscillation amplitude level. Figure 61 shows that the highest oscillation amplitude level $\widetilde{V}$ is obtained for higher d.c. currents and magnetic fields between 300 mT and 500 mT and for d.c. currents between 7mA and 9mA. these properties are directly linked to the sample properties. During the reservoir computing experiments, we use the non-linearity of the oscillation amplitude with the d.c. current to compute. In practice, we fix the magnetic field and we vary the current that receives the sample. The next figure represents this non linear dependency for 3 different magnetic fields.

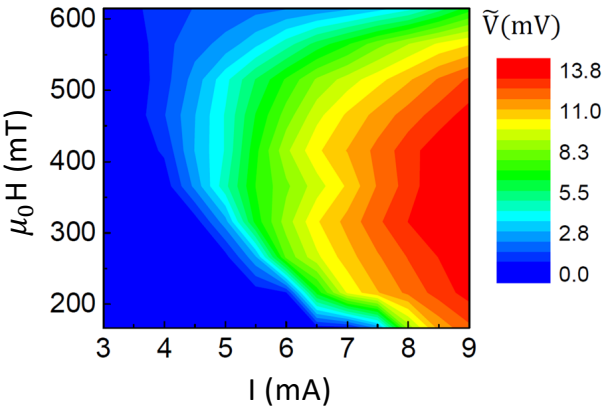

Figure 61: Amplitude Voltage $\widetilde{V}$ of the oscillator in the steady state: map in the I-H plane.

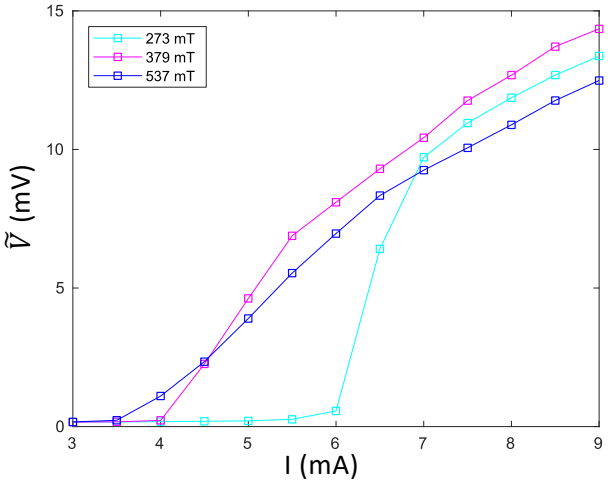

Figure 62: Amplitude Voltage $\widetilde{V}$ of the oscillator in the steady state as a function of I for $H = 273$ mT (cyan), $H = 379$ mT (magenta) and $H = 537$ mT (blue).

As it can be observed in Figure 62, the magnetic field modifies the threshold current. For 297 mT this threshold current is around 5.9 mA and for 379 mT and 537 mT it is more around 4.0-4.4mA. This observation can be also made on the 2D map. The threshold current corresponds to the transition between dark blue (no emission) and colored areas. Increasing the magnetic field decreases the threshold current because less energy is required to compensate the damping

(see part 2). On the 2D map we can observe that the threshold current decreases rapidly from 200 mT to 400 mT and then much more slowly from 400 mT to 600 mT. As it can be observed in the figure, besides changing the threshold current, the magnetic field seems to change the general shape of the non-linearity, it seems to "stretch" it on the direction of d.c. current. The amplitude of the oscillation is directly linked to the vortex orbit. We remind that the oscillation amplitude is given theoretically by the following equation:

$$\widetilde{V}(t) = \lambda(H_{\perp}, I)s(t) \tag{64}$$

Where s is the vortex radius and λ is a factor depending on the perpendicular magnetic field $H_{\perp}$ and the d.c. current. s is a function of I/I_{th} (as seen in section 4.2.2 it evolves more or less like $\sqrt{((I/I_{th} - 1)))}$. So the stretching effect is due to the fact that when I_{th} decreases, the range of I/I_{th} increases. As seen in section 4.2.2, the critical current decreases with the magnetic field. The fact that $\widetilde{V} > 0$ before I/I_{th}=1 for high magnetic field is due to the thermal fluctuations [157]. The oscillation amplitude as a function of I/I_{th} is presented in next Figure 63. At the end best performances are obtained for

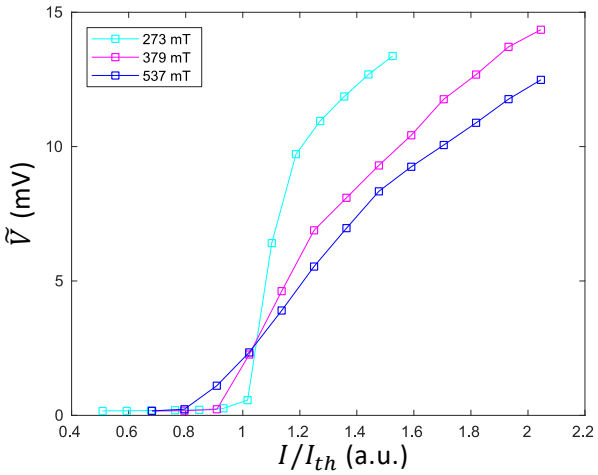

Figure 63: Amplitude Voltage $\widetilde{V}$ of the oscillator in the steady state as a function of normalized current I/I_{th} for $H = 273$ mT (cyan), $H = 379$ mT (magenta) and $H = 537$ mT (blue).

intermediate magnetic fields where the transition is not too brutal (in contrary to low fields) nor too sharp. Now that influence of the

magnetic field on the non-linearity used for computation is explained, we focus on the role of the d.c. current to see how it influences the amplitude and the asymmetry of the measured time traces. Figure 64

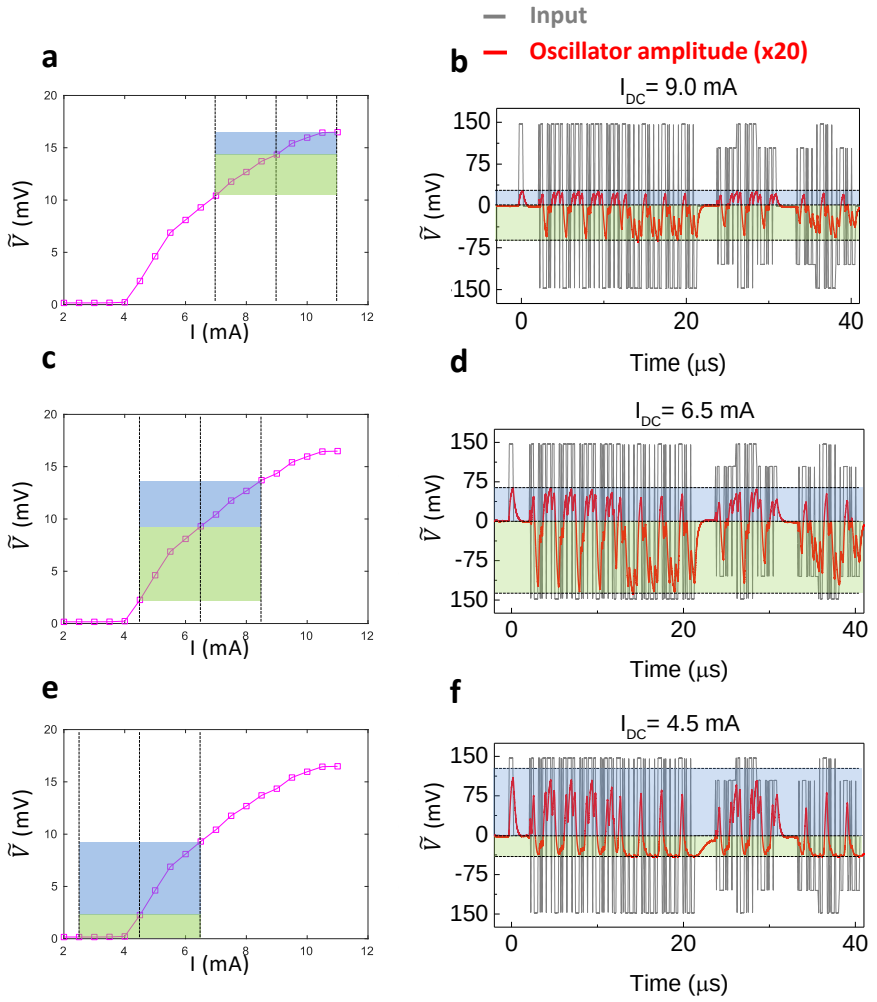

Figure 64: Influence of the d.c. current on the oscillation amplitude variation: a,c,e $\widetilde{V}$ as a function of I. The blue (green) shaded area highlights the typical excursion in the voltage amplitude V_{up} (V_{dw}) that results when an input signal of $V_{in} = +150$V ($V_{in} = -150$ mV) is injected. b,d,f The oscillator voltage amplitude, curves (in red) in response to the input waveform (in gray). Blue (green) shaded area represents V_{up} (V_{dw}). Curves are plotted for $H = 379$ mT and $I = 9.0$ mA (a,b), $I = 6.5$ mA (c,d) and $I = 4.5$ mA (e,f).

focuses on three of the different d.c. currents used in section 8.3.1

which are 4.5 mA, 6.5 mA and 9.0mA. On the left part of the figure, the non linear dependence of the oscillation amplitude with the current is represented. The part of this non-linear dependence explored when the input is sent is represented in colored area. An input of 300 mV induces a typical variation of d.c. current of coarsely 4 mA. So when the input is sent, the oscillator receives a current range of ±2mA around the fix d.c. current. By changing the d.c. current, one explores different windows of the non linear dependence of $\widetilde{V}$ with I which is set by the magnetic field and the larger is the input amplitude, the larger is the explored window. This exploration window is represented by the vertical dashed lines. The central line is the d.c. current and the left and right lines are the extreme current values received by the oscillator. The level of the oscillation amplitude when no input signal is sent is the intersection of the central vertical line and the magenta curve $\widetilde{V} = f(I)$. The higher oscillation amplitude value reached is the intersection of the right vertical line and the curve $\widetilde{V} = f(I)$. V_{up} the difference between the highest amplitude reached by the oscillator amplitude when input is sent and the oscillation amplitude level due to the d.c. current is represented in blue. Similarly the lowest amplitude reached when the input is sent is situated at the intersection between the left vertical line and the $\widetilde{V} = f(I)$ curve. V_{dw} the difference between the oscillation amplitude level due to the d.c. current and the lowest amplitude reached by the oscillator amplitude when input is sent is represented in green. On the right part of the figure, the time traces corresponding to these different d.c. current conditions are plotted (it is the same time traces as in Figure 58). V_{up} and V_{dw} are also visualized on time traces with the blue and green areas. We can see that the apparent asymmetry of the measured time traces and the amplitude variation is completely determined by the portion of the non-linearity $\widetilde{V} = f(I)$ which is explored. For 9.0 mA d.c. current value, the signal has a rather small amplitude because the oscillation amplitude evolves in a part of the non-linearity $\widetilde{V} = f(I)$ where variations are small (for high current the growth of $\widetilde{V}$ with I slows down). For 6.5 mA, the lowest current received by the oscillator is close but still superior to the threshold current (the oscillator receives currents between 4.5 mA and 8.5 mA approximatively). The region close to the threshold current is a region of important variation. The largest oscillation amplitude variations are thus obtained in such

conditions. The overall amplitude of the time traces for 6.5 mA is much larger than for 9.0 mA d.c. current value. Because the variation of the oscillation amplitude is stronger close to the threshold current, the V_{dw} (green) is much larger than the V_{up} (blue). Decreasing the d.c. current to 4.5 mA the lowest d.c. current that receive the oscillator is around 2.5 mA (under the threshold current at 4 mA). Saturation occurs thus in the time traces (see green area) which is detrimental for computation. All the current input values below the threshold current give the same oscillation amplitude 0. So these inputs in later computation can not be differentiated. In such cases the asymmetry of the time traces is reversed with V_{up} which is larger than V_{dw}. As a conclusion of this section, to have a satisfactory oscillation amplitude variation, one should choose an intermediary magnetic field (typically between 300 mT and 500 mT) so the variation of the $\widetilde{V} = f(I)$ non-linearity is not too abrupt such as for low field (because of high I_{th} value) nor too slow such as for high fields. Once the magnetic field is fixed, in order to obtain large variations of the oscillation amplitude, the d.c. current should be chosen such as when the oscillator receives the AWG signal, the lowest current it receives is close to the threshold current.

8.3.4 *Noise and non-linearity trade-off*

A good classification performance requires non-linearity in the reservoir as it has been seen in chapter 5. Indeed the non-linearity will allow the separation of the different inputs when the initial problem is projected in higher dimension in the reservoir state. A good signal to noise ratio is also important, because if it is not the case similar inputs will result in very different reservoir states because of the noise and thus the reservoir will fail in the approximation property. The non-linearity and the voltage noise vary considerably with d.c. current and magnetic field as it can be seen in Figures 65. To quantify the non-linearity, we computed the second derivative of the voltage with the current $\partial^2 \widetilde{V} / \partial I^2$. As it can be observed in Figure 65a, the highest non-linearities are close to the threshold current where the emission of the oscillator varies strongly with the current. The noise level is evaluated taking the standard deviation of the fluctuation due to the noise (similarly to the noise evaluated in chapter 7 Figure 50). As it

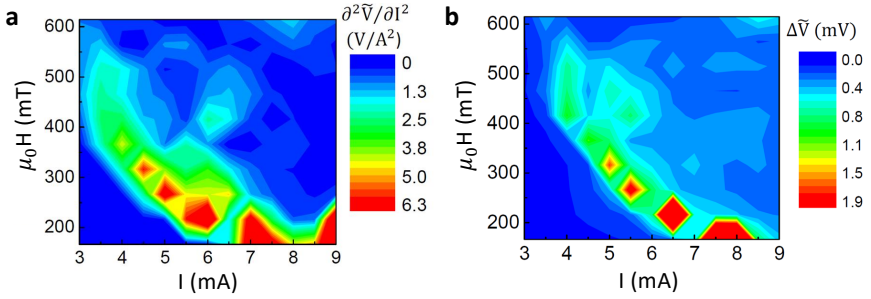

Figure 65: a. The non-linearity $\partial^2\widetilde{V}/\partial I^2$ of the oscillator: map in the I-H plane. b. Amplitude noise $\Delta\widetilde{V}$ of the oscillator in the steady state: map in the I-H plane.

can be observed in Figure 65b, the noise also varies strongly close to the threshold current.

Since spin-torque oscillators have a small magnetic volume, thermal noise affects the magnetization dynamics. The resulting voltage amplitude noise is large for large non-linearity, which quantifies the sensitivity of the system to perturbations. The correlation between voltage noise and non-linearity appears clearly in the comparison of Figure 65a and b. Neuron non-linearity is a key ingredient for classification as it allows the separation of input data. On the other hand, noise in neuron response is detrimental for classification as it directly affects the output. Figure 57 shows the classification performance as a function of d.c. current and field. We find good performance by choosing a bias point with intermediate non-linearity and therefore intermediate noise, and where the neuron output changes strongly in response to the ac input. Such bias points allow enough non-linearity to classify while keeping large enough signal to noise ratio to distinguish between outputs.

8.4 CONCLUSION

We have seen in this chapter that for experimental reservoir computing with a spin-torque oscillator the θ time scale and the operating point could be optimized. The θ time step plays both an important role in the memory of the reservoir (particularly when intrinsic memory is the only memory mechanism) and in the connectivity between the vir-

tual neurons. A trade off should be found to have a θ short enough to assure memory and connectivity and long enough to assure a satisfying amplitude variation. Best trade-off is found for $\theta = T_{relax}/2$ where T_{relax} is the relaxation time of the oscillator. Adding a numerical shift suppresses the need of memory and results are dominated by the non-linearity and a satisfying signal to noise ratio. The operating point plays an important role in order to optimize these properties. First the magnetic field should be intermediary (between 300 mT and 500 mT) so the variation of the non-linearity $\widetilde{V} = f(I)$ is not too abrupt nor too smooth. To obtain a large signal amplitude, at a given field the d.c. current should be big enough to avoid the saturation due to reaching the threshold current and low enough to avoid a regime where $\widetilde{V}$ variation saturates. More generally, the operating point should be chosen at intermediate non-linearity and noise regime which mean not too close nor too far from the threshold current.

9

CRITICAL ROLE OF THE NON-LINEARITY IN THE DATA PROCESSING ON SPEECH RECOGNITION TASKS

In this chapter, we investigate the effect of the time-to-frequency filtering method used for spoken digit recognition, since the choice of frequency filtering influences strongly the final recognition rate (see section 7.2.3). The mel-frequency cepstral coefficients (MFCC) and the Lyon's cochlear model are the most common methods used in speech recognition. Both methods mimic the time-to-frequency transformation occurring in the biology [260, 261]. In the previous chapter 7, a simpler filtering method called spectrogram was also used for comparison. The first section will describe these three methods and analyze the performance of such methods as stand-alone feature extractors. The second section shades light on the critical role of non-linearity in the filtering method. The last section analyzes the gain obtained when a spin-torque oscillator is added to the computation.

9.1 MFCC AND COCHLEAR DECOMPOSITION AS STAND ALONE FEATURE EXTRACTORS

The cochlear decomposition was first used for recognizing digits with reservoir systems in [234] and since then, it became frequently used to benchmark reservoir computing approaches [229]. Notably it has been widely employed for hardware reservoir computing studies [239, 244, 246, 245, 251, 247, 252, 2, 250, 248], while MFCC is commonly used for speech recognition in software applications [262]. It is important to note that both of these methods were developed before machine learning became popular, and thus were designed to extract the useful features of an audio signal, without any learning aspect.

9.1.1 *The Mel Frequency Cepstral Coefficient method (MFCC)*

The MFCC method was developed in the eighties in order to obtain a relevant parametric representation of acoustic signal for recognizing monosyllabic words [261]. The goal is to compress the speech data by eliminating non pertinent information for phonetic recognition and enhancing the aspects relevant for phonetic distinction. The first step is to perform a fast Fourier transform (FFT) of the signal to process. The signal is sampled in our case at the rate of 12.5 kHz and windows of 6.4 ms (80 samples) are used for the FFT. The Fourier spectra is filtered through 20 different triangular functions, spaced following the logarithmic mel scale (Figure 66) [263].

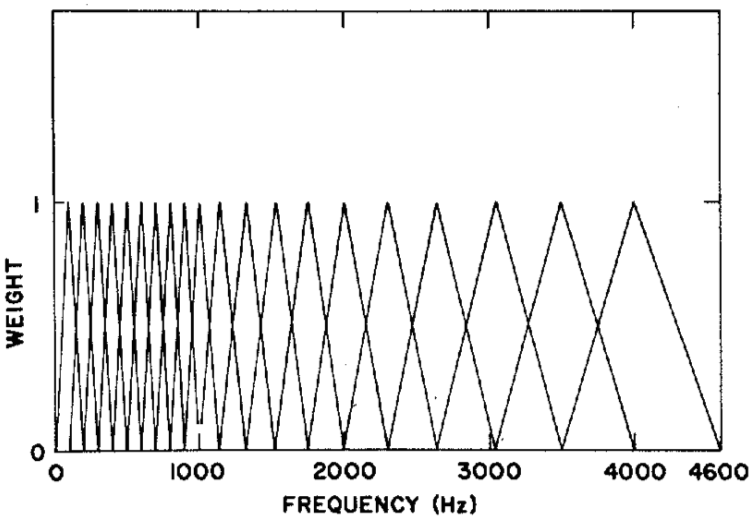

Figure 66: Filters for generating mel-frequency cepstrum coefficients. Reproduced from Davis et al 1980 [261].

This psycho-acoustic scale mimics the human perception of pitches [263]. Finally the MFCC are computed as:

$$MFCC_i = \sum_{k=1}^{20} E_k cos\left[i\left(k - \frac{1}{2}\right)\frac{\pi}{20}\right] \tag{65}$$

In equation 65, E_k is the log-energy of the k^{th} filter output and i is the index of the cepstral coefficient. In our case, we use the classical auditory toolbox from Slaney [264], where by default $i =$

$1, 2, ..., 13$. During this filtering process, the signal undergoes non-linear transformations, when computing E_k, the log-energy of the filter output.

9.1.2 *The cochlear decomposition method*

The Lyon cochlea model was also developed in the early eighties [260]. Compared to the MFCC method, this model is much closer to the biology of the human cochlea. The human cochlea maps the different frequencies of a signal to the spatial domain, because each of its sections is more sensitive to a particular range of frequency. Figure 67 represents the general principal of the Lyon cochlea model. The sound pressure wave is first filtered, then the resulting band-passed signals are detected through half wave rectifiers (HWR) and compressed by interconnected automatic gain control (AGC). The

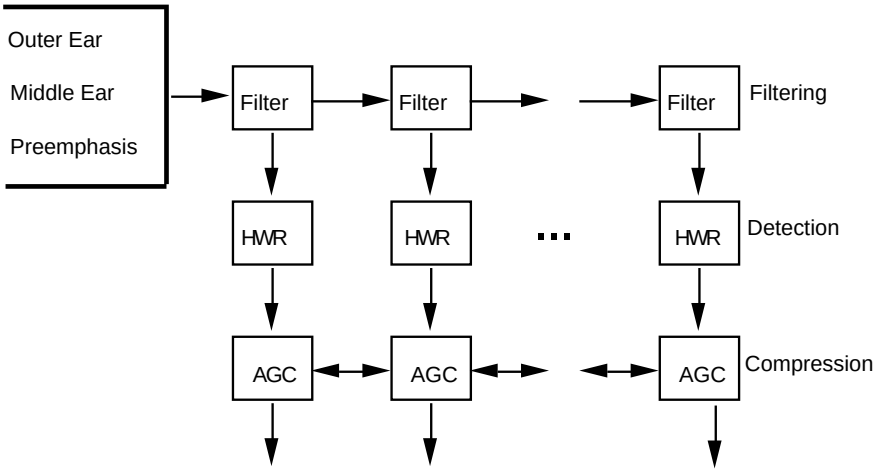

Figure 67: Structure of Lyon cochlea model: the pressure waves enter the cochlea and undergo a series of filters. The filtered signals are detected by half wave rectifiers (HWR). The rectified signals are compressed by interconnected automatic gain controls (AGC). Reproduced from Slaney et al 1988 [265].

traveling of pressure waves through a cochlea section is modeled by a notch filter. The sensitivity of each section of the cochlea is modeled by a resonator centered on a frequency specific to this section. The combination of these two filters selects only a narrow band of frequency

(Figure 68). These filters are the equivalent of the triangular filter in

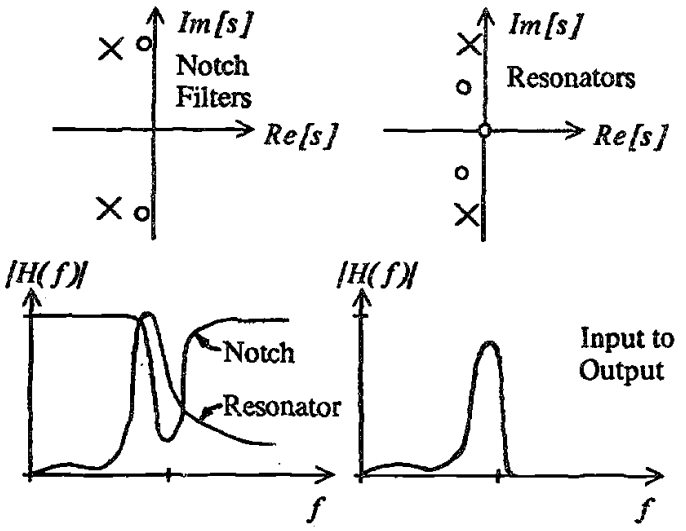

Figure 68: Pole-zero plots and transfer function of the notch filters and resonators, used in Lyon cochlea model. Reproduced from Lyon et al 1982 [260].

the MFCC method. Then the bandpass signal is detected by a half wave detector and compressed by an AGC, that model respectively the behavior of the hair cells and the masking of the ear [266]. The non-linearity occurs in these two last steps (Figure 69). The first non-

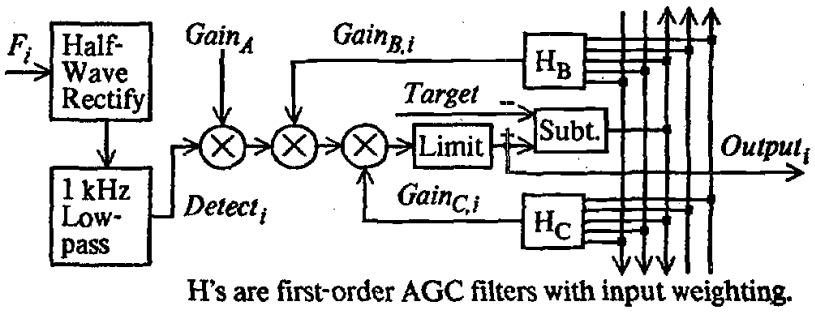

Figure 69: Block diagram of one channel of the detection and compression models. Extracted from Lyon et al 1982 [260].

linearity comes from the half-wave rectifier. Then the AGC computes gains, that multiply the rectified signal, so in the long term the channel values converge toward a target value. This AGC reproduces the masking of human ear. For a given channel, the different gains of the AGC depends on the gain values from the neighboring channels. For these reasons, the AGCs can be seen as interconnected non-linear units. Moreover the gains are updated depending of their previous values, and thus the AGCs introduce memory.

9.1.3 *Spectrogram method*

The spectrogram method consists in taking the real part of the fast Fourier transform (FFT), applied to the audio signal. This preprocessing was chosen in [2] to extract the frequency information from a signal without introducing non-linearity. Indeed the Fourier transform and the real part function are both linear operators. In practice, the FFT is performed on Hamming windows [267] of 128 samples (10.2 ms), that are overlapping on 124 samples. Each FFT thus returns 65 complex coefficients and the real part of them is kept. The resulting matrix of spectra is then decimated by 20 (we conserve only one spectra in 20), in order to reduce the size of the matrices representing the digits. Compared with previous methods, this transformation does not involve non-linear transformations after the FFT.

9.1.4 *The MFCC and the cochlear decomposition results on spoken digit recognition*

The MFCC and the cochlear decomposition can achieve by themselves very high recognition levels: just training a linear classifier on their frequency channels allows reaching up to 95.8% and 77.2% word success rate (WSR) (Figure 70). On the other hand, as observed previously in chapter 7, the linear spectrogram leads to 10% WSR, which is consistent with a random choice level. If training a linear classifier on the channels leads to good classification results, the speech recognition problem is already mainly linearly separable. The MFCC and cochleagrams (matrices resulting from the Lyon's cochlea model) already hold in their frequency channels the features

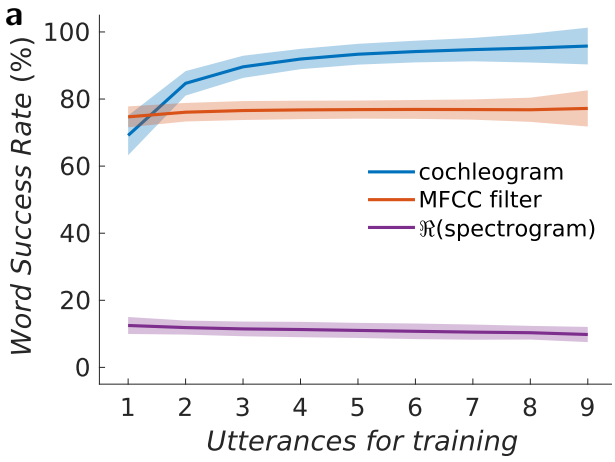

Figure 70: Spoken digit recognition rates as a function of the number of utterances N used for training of the filtered input (without neural network) corresponding to three different methods: cochleogram, MFCC filter and linear spectrogram. The shaded area corresponds to the incertitude on the WSR due to the different possible choices of training sets and test sets.

needed to recognize the different digits, whereas in the case of the spectrogram, these features remain yet to be extracted. Indeed, in the case of the MFCCs and cochleagram, frequency filtering already achieves the goal of a reservoir network, because the initial problem is non-linearly projected into a higher dimension state. The main difference compared with a reservoir network is that these non-linear projections are not random, but they are designed by hand to mimic the biology or the human perception of sounds. Our hypothesis is that the origin of the good recognition rate, observed earlier, would be the presence of a non-linear transformation in the filtering and that many kind of non-linearities, not necessarily inspired from the biology of the ear, can achieve similar results. In order to investigate the role of non-linearity for data separation, in the next section, we will evaluate the effect of a simple non-linear exponent on the spectrogram to separate the digits.

9.2.1 *Principle of the analysis*

After the spectrogram filtering, each word, indexed by γ (with $\gamma \in [1, 500]$, because there are 500 words in the TI46 spoken digit data base we use), is projected in several vectors of 65 frequency channels. Let's call Z_γ the real part of the resulting matrix of dimension $m \times p_\gamma$, where $m = 65$ is the number of channels and p_γ is the length of the matrix associated to the digit γ. Before applying the non-linearity, each Z_γ matrix is normalized by the maximum of it absolute value, giving the $\widehat{Z}_\gamma$ matrix.

$$\widehat{Z}_\gamma = \frac{Z_\gamma}{\max(|Z_\gamma|)} \tag{66}$$

This normalization avoids having values that become too large to be computed numerically. We normalized per word and not over all the words in the data base, because in a real situation, the only accessible information is the word to recognize. The following equation summarizes how the non-linearity is applied.

$$S_\gamma = \Re((\widehat{Z}_\gamma)^\alpha) \tag{67}$$

Importantly, S_γ is designed such that the only non-linearity comes from the α factor. The other operators (real part) are linear. For $\alpha = 1$, the filtering $\Re(spectrogram)$ is retrieved.

9.2.2 *Results for various non-linearity levels*

After computing S_γ for various α factors and for all the spoken digits data base, a linear classifier is trained on the channels of S_γ for 90% of the digits and tested on the remaining 10%. The resulting word success rate (WSR) for testing as a function of the alpha factor is shown in Figure 71. The shaded area represents the incertitude coming from the different possible training sets. Two tendencies seems to superpose. Generally when α increases, the WSR tends to decrease. In addition, the WSR seems to drop in the vicinity of odd values of α. For $\alpha = 1$, the WSR of 10% corresponding to random classification is

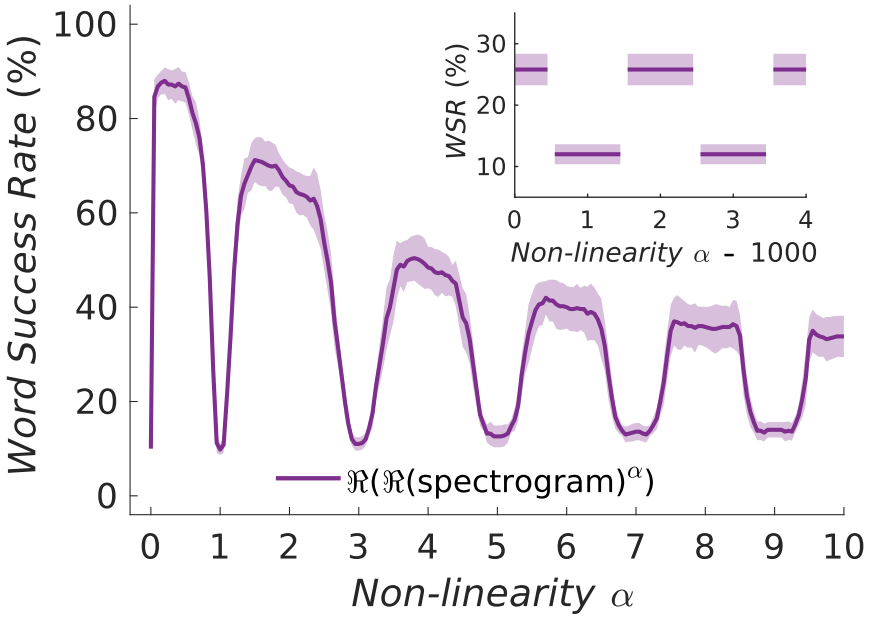

Figure 71: Word success rate (WSR) as a function of the non linear exponent α of the spectrogram real part (Inset: WSR for large non-linearity coefficient values, $\alpha = 1000 - 1004$). The colored region corresponds to the standard deviation of the recognition rate, based on training with all possible combinations.

retrieved. The case $\alpha = 0$ is also trivial, because all the channel of the S_γ are equal to 1, so digits cannot be classified. For $0 < \alpha < 1$, the WSR can reach very high values: up to 88% for $\alpha = 0.2$. This WSR is higher than the one obtained for MFCC method. For $1000 < \alpha < 1004$, the WSR shifts between a low values (12%) around odd values of α and a higher value (25.8%) around even values of α.

Since $\Re(spectrogram)$ is renormalized for each digit γ, all the values of S_γ are between $]-1, 1[$ except for one value per digit which is either equal to 1 or -1. When α becomes large, the values of S_γ become smaller and smaller and differences between the different digits γ fade, explaining the general decrease of the WSR when α increases. In the extreme case of $\alpha \to \infty$, all the values of S_γ are equal to 0, beside one value per S_γ which is either equal to 1 or -1.

The apparent difference between odd and even valued numbers is explained by the sign of the values of S_γ: if S_γ takes only positive values, the WSR is better than if it takes both positive and negative values.

Let's call $\widehat{Z}_\gamma^-(i,j)$ and $\widehat{Z}_\gamma^+(i,j)$ respectively the negative and positive values of $\widehat{Z}_\gamma$. Let's decompose $\alpha = n + \epsilon$, $n \in \mathbb{N}$ and $\epsilon \in]-0.5, 0.5[$. The value $S_\gamma^-(i,j)$ and $S_\gamma^+(i,j)$ corresponding to $\widehat{Z}_\gamma^-(i,j)$ and $\widehat{Z}_\gamma^+(i,j)$ are given by the equation:

$$\begin{cases} S_\gamma^-(i,j) = |\widehat{Z}_\gamma^-(i,j)|^{n+\epsilon}(-1)^n cos(\pi\epsilon) \\ S_\gamma^+(i,j) = |\widehat{Z}_\gamma^+(i,j)|^{n+\epsilon} \end{cases} \tag{68}$$

When n is even, all the values of S_γ are positive, on the contrary when n is odd, $S_\gamma^-(i,j)$ are negative and $S_\gamma^+(i,j)$ are positive. Apparently in the particular case of speech recognition, the sign of the real part of the spectrogram $\widehat{Z}_\gamma$ is not a relevant information and suppressing the negative sign helps recognition. For $\alpha \to \infty$ and $\alpha \in]2n - 0.5, 2n + 0.5[$, each digit possesses one value of S_γ which is not zero. Learning the index (i,j) of this unique positive number allows identifying the correct digits in 25.8% of the cases. On the contrary for $\alpha \in]2n + 0.5, 2n + 1.5[$, the sign blurs the information of the unique non zero value and WSR decreases to 12%.
To conclude, a simple spectrogram with a non-linearity exponent α showed similar performances for $0 < \alpha < 1$ than the more complex MFCC and cochlea decomposition method. These results stress the crucial role of non-linearity. The chosen non-linearity should not be completely random to give good performance, because it should lead to positive values independently of the sign of the spectrogram real part and should not compress too much the data (e.g., $\alpha \to 0$ or $\alpha \to \infty$).

9.2.3 *Separation property of the different filters*

In order to summarize the separation property of each filtering method, we visualize how they map the different digits in a two dimension map. Each filtering method represents each digit γ by a matrix of frequency channels, that we call S_γ by analogy with the previous section. To simplify the representation, we condense these matrices into average vectors over the columns, called R_γ. The values of R_γ are defined as follows:

$$R_\gamma(i) = \frac{1}{p_\gamma} \sum_{j=1}^{j=p_\gamma} S_\gamma(i,j) \tag{69}$$

Each vector R_γ is an input to classify, with a dimension $m \times 1$. m represents, as before, the number of frequency channels which is 13 for the MFCC method, 78 for the cochlear decomposition method and 65 for the spectrograms (with and without the non-linear factor). The t-SNE transformation [268] is a non-linear projection from a n-dimensional space to a 2-dimensional space, which conserves the probability for two different points to be neighbor. A t-SNE transformation is performed on the R_γ vectors, in order to visualize in 2D map if the R_γ vectors, for each class of digit, are separated. Figure 72 plots the two components of the t-SNE for the distribution of R_γ vectors. Each color represents

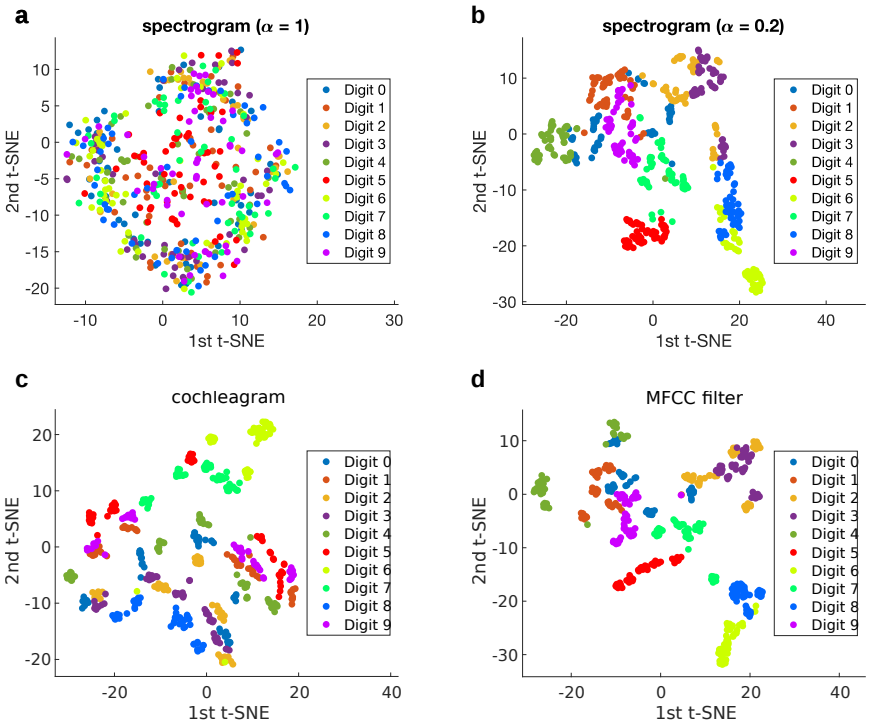

Figure 72: 2D representation of the two t-SNE components for: a, the spectrogram with $\alpha = 1$, b, the spectrogram with $\alpha = 0.2$, c, the cochleogram, and d, the MFCC filtering methods.

the R_γ for a given class of digit. For the spectrogram (Figure 72a), the different R_γ vectors do not form separate clusters, depending on the class of digit they belong to. This result, consistent with the fact that the problem is not linearly separable, explains the poor word success rate which was obtained. In Figure 72b, the best classification

situation for spectrogram with $\alpha = 0.2$ is plotted. In this case, the depending on their class, the digits form separate clusters, which supports that a linear classifier can distinguish correctly 88% of the digits. In Figure 72c, for the cochleagram, the data is well separated because neighboring R belong to same class. For the cochleagram, the transformation is more complex than for the spectrogram with α= 0.2 and the R vectors from one class form several clusters. Finally in Figure 72 d, for MFCC, the digits are well separated and the data distribution visualized with the t-SNE method is similar to the case in Figure 72b.

The t-SNE visualization illustrates well the separation that occurs among the data, depending on the time-to-frequency transformation. In particular, it shows that a spectrogram with a non-linear factor can achieve similar separation to the more complex MFCCs and cochlear decomposition.

9.3 Gain in recognition rate brought by an oscillator used in the framework of reservoir computing

After testing the filtering as a stand alone feature extractor, we add an oscillator to implement reservoir computing after the filtering, similarly to section 7.2.2. We quantify the gain in word success rate that an oscillator brings after each kind of filtering. Indeed, by quantifying the gain in WSR, we quantify the net contribution of the oscillator to the recognition. A hardware component is shown to be promising for reservoir computing only if it provides a positive gain, when compared with the filtering as a stand alone feature extractor. We first simulate the behavior of a quite general non-linear node with relaxation. We compare later these results with experimental results using a spin-torque nano-oscillator.

9.3.1 *Simple oscillator model*

We have developed a simple model based on a non- linear spintronic oscillator [157] taking into account the main ingredients for neuromorphic computing: non-linearity (root square dependence on the

input current) and memory (relaxation time of the oscillator between two different emitted voltage levels) [153, 2]. The dynamics of the evolution of the oscillator's emitted voltage amplitude $\widetilde{V}_i$ as a function of the input voltage V_i^{in} at time step i can be numerically solved as follows:

$$\widetilde{V}_i = \widetilde{V}_{i-1}^{\infty} e^{-\Delta t/T_{relax}} + \widetilde{V}_i^{\infty}(1 - e^{-\Delta t/T_{relax}}) \tag{70}$$

where T_{relax} is the relaxation time towards the asymptotic value $\widetilde{V}_i^{\infty}$ given, similarly to section 4.2.2, by:

$$\widetilde{V}_i^{\infty} = C\sqrt{I_{DC} - (V_i^{in}/R) - I_{th})} \tag{71}$$

with C, a constant related to the initial bias condition, i.e. the initial emitted voltage of oscillator, R the DC resistance of the oscillator and I_{th} the threshold current above which auto-oscillations can occur. In order to simulate the oscillator's response to a time varying input (V_i^{in}), equation 70 is numerically solved using the following parameters: $\Delta t = 5$ ns, $V_{in}^i/R = \pm 3$ mA, $I_{DC} = 4.9$ mA, $T_{relax} = 410$ ns. Note that these parameters are mainly extracted from experiments as reported elsewhere [2]. This model is quite general and can be adapted to other systems by changing T_{relax} and $\widetilde{V}_i^{\infty}$. Indeed Appeltant et al used a similar model for Mackey-Glass and Ikeda non-linear nodes [239, 257]. In this later case, the main point that differs is the type of non-linearity introduced in the model. Compared with a real spin-torque oscillator, one of the main limit of this model is that it does not take into account the variation of the relaxation time with the current (see section 4.2.1 and 4.2.2). Moreover it does not describe well the behavior of $\widetilde{V}$ when its value is very far from $\widetilde{V}^{\infty}$. In the next section, this simple model is compared with experimental results, in order to test its validity.

9.3.2 *Gain for different methods of filtering*

The gain of recognition rate induced by the emulated non-linear oscillator is computed in Figure 73a. The recognition gain of the emulated neural network is obtained by subtracting the contribution from acoustic filters (previously calculated in Figure 70) to the total recognition rate. The gain provided by the non-linear oscillator

is low for both non-linear filters: up to 4-9.4% (depending on the number of utterances used for training) for the cochlear method and 22% for MFCC method. It is because these filters already perform well as stand alone feature extractors, so there is not much room for improvement with an additional neural network (here the oscillator used as a reservoir computing network), even though the final WSR is close to perfect. On the contrary, after the spectrogram, the neural network improves drastically the recognition gain up to 60.2% (over 9 utterances trained). Finally, the simulations were compared

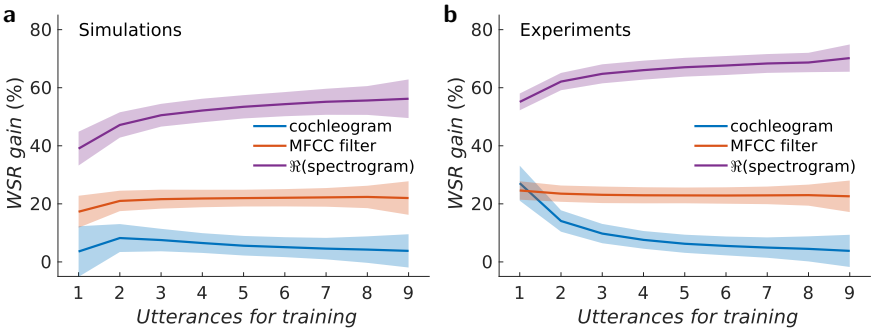

Figure 73: a. Gain of WSR as a function of the number of utterances N used during training for a non-linear oscillator modeled according to equation 70-71. b. Gain of WSR for the experimental reservoir with a time-multiplexed spin torque nano-oscillator driven by spin polarized current. The colored region corresponds to the standard deviation of the recognition rate, based on training with all possible combinations.

with an experimental non-linear oscillator. The gain on the spoken digit recognition for the three different acoustic filters induced by the experimental spin torque nano-oscillator is shown in Figure 73b. Despite the simplicity of the model, a good agreement between the experimental results and the simulations can be observed. The gain varies between 4 and 27% for the cochleogram, 22 and 24% for the MFCC filter, and between 55 and 70% for the linear spectrogram depending on the number of utterances used during the learning process. For some cases, notably the linear spectrogram, the experimental spin-torque nano-oscillator exhibits slightly higher recognition gain than the simulations even if the latter does not consider the intrinsic noise. This is mainly due to the higher complexity in the dynamics

of the spin-torque nano-oscillators compared to the model, including current-dependent relaxation times [157, 153]. Such higher complexity further enhances the spoken digit recognition performance. In order to evaluate the net contribution of the oscillator in the recognition, the gain of WSR when compared with the filtering as stand-alone feature extractor should be computed. Both using simulations and experimental spin-torque oscillator, we obtained high gain of 60% in simulation and 70% experimentally for simple linear spectrogram method and lower gain of 22% and 4% for MFCC and cochlear decomposition. These results shows that the gain in WSR, as the final WSR, depends strongly of the filtering method chosen. Therefore the conclusions regarding the efficiency of a hardware for spoken digit recognition should be drawn regarding to the filtering method used and the gain in WSR the hardware provides.

9.4 CONCLUSION

Different frequency filtering methods were tested as stand-alone feature extractors. Training a linear classifier on the S_γ matrix for the TI-46 spoken digit data base [269], for cochleagram and MFCC gives high word success rate, which indicates that the different classes of digits were mainly separated by the frequency filtering. On the contrary, the real part of a spectrogram does not separate the inputs depending on the digit class.

By adding a non-linearity to the spectrogram, we show that we can reach similar results, which stresses that the reason of separation in MFCC and cochlea decomposition is the presence of non-linearity, but the type of non-linearity is not that important. Indeed we reached similar performances just applying a power exponent to the spectrogram which is much simpler than the non-linearity occurring in the cochlear decomposition or in the MFCC methods.

In a second part, a non-linear oscillator is added to process the filtering. The gain in WSR brought by the non-linear oscillator is computed for each filtering method. The non-linear oscillator was first simulated, and these results were compared with experimental results, showing good agreement. For the non-linear methods MFCC and cochleagram, the gain of WSR is small, despite a nearly perfect WSR. On the contrary with the linear spectrogram, the gain of WSR is much higher

even though the final WSR is more around 70-80%.
The final WSR should thus be interpreted with caution, depending on which time-to-frequency transformation is used. When such task is performed to prove the efficiency of a type of hardware system for computing, the demonstration is done only if the hardware provides a gain of WSR. In order to test and compare hardware systems, using a linear spectrogram eases the interpretation of the results, because it does not introduce any separation of the word to recognize prior the hardware system.

Part V

IMPROVED RECOGNITION THROUGH DELAYED FEEDBACK MEMORY

10

EXPERIMENTAL IMPLEMENTATION AND EVIDENCE OF FADING MEMORY

The following chapter presents how extrinsic memory was experimentally added to a spin torque oscillator. In the first section, we measure the intrinsic memory of the oscillator in order to quantify its extension. In the second section the experimental implementation of the delay feedback is presented. In the third section, the range of fading memory brought by the delayed feedback is evaluated.

10.1 EVALUATION OF THE INTRINSIC OSCILLATOR MEMORY

10.1.1 *Principe of the relaxation time measurement*

In this section we explain how we extract the relaxation time depending on the operating point. When a d.c. current and a magnetic field are applied to an oscillator, the vortex core in the free layer gyrates on a stable orbit. The idea of our measurement is to slightly distort the trajectory of the vortex core by applying an electrical pulse. The electrical pulse induces an additional spin-torque on the vortex core, so it reaches another orbit. After the end of the electrical pulse, the vortex core returns progressively to its initial stable orbit imposed by the operating point. We evaluate the time the vortex core takes to come back to its stable orbit, to measure how long the spin torque oscillator keeps traces of the electrical pulse perturbation. This time is evaluated when the perturbation increases the radius of the vortex position (the spin torque induced by the electrical spike works against the damping) and when the perturbation decreases the radius of the vortex core position (the spin torque induced by the electrical spikes reinforces the damping). To evaluate this time, we measure the amplitude of the oscillating voltage emitted by the oscillator which is directly proportional to the vortex core radius [153]. Interestingly, because the measure of the relaxation time is achieved after the end of the spike voltage spike, it does not suffer from errors due to parasitic

capacitance in the circuit or due to relaxation time of the applied d.c. current.

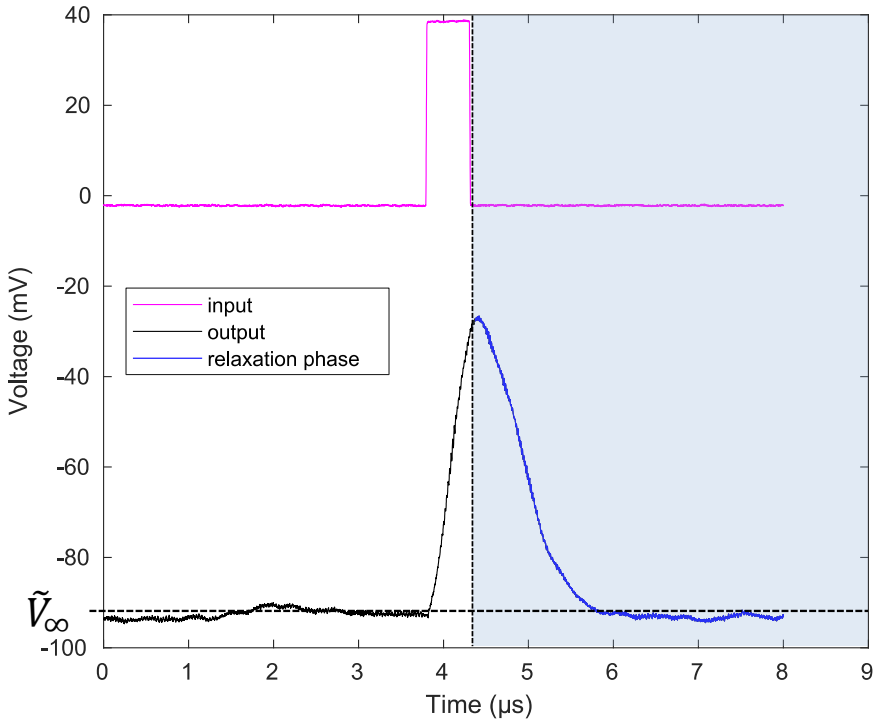

Figure 74: Measurement of the oscillation relaxation. Upper graph: signal (magenta) sent by the arbitrary waveform generator. Lower graph (black and blue): output signal after the diode (oscillation amplitude $\widetilde{V}$ after amplification by 20 dB). After the end of the voltage pulse, $\widetilde{V}$ returns to it initial level $\widetilde{V}_\infty$. This relaxation phase is highlighted in the transparent blue area. The operating point is H=300 mT and I=-5.5 mA.

In practice the measurement setup is the same as the one described in Figures 42 and 43. The input generator sends a 0.4 μs pulse of $\pm$40 mV which corresponds approximatively to a $\pm$1 mA variation of the d.c. current received by the oscillator (Figure 74 magenta in the case of a positive spike). The oscillation amplitude $\widetilde{V}$, after amplification, is measured (Figure 74 black and blue curve). The relaxation time T_{relax} is extracted from the relaxation using an exponential fit on a part of the relaxation phase time trace (Figure 74 blue curve). In the

auto-oscillator model [157] the dynamics of the oscillator amplitude can be indeed described by the normalized amplitude u:

$$u(t) = \frac{\widetilde{V}(t)}{\widetilde{V}_{\infty}} \tag{72}$$

where $\widetilde{V}$ is the oscillation amplitude and $\widetilde{V}_{\infty}$ is the oscillation amplitude in the steady state. u should evolve as [157]:

$$\frac{du}{dt} = \Gamma_P(1 - f(u)) \tag{73}$$

with Γ_P the oscillator relaxation constant and f a non-linear function. If $\widetilde{V}$ is close to $\widetilde{V}_{\infty}$, $f(u) \approx f(1)$. In this case, the solution to equation 73 is an exponential with decay time:

$$T_{relax} \approx \frac{1}{\Gamma_P(1 - f(1))} \tag{74}$$

T_{relax} quantifies how long the information is conserved. In order to stay in a situation where $u \approx 1$ and where an exponential fit of $\widetilde{V}$ is valid, the values of $\widetilde{V}$ selected to be fitted should respect approximately $\widetilde{V}_{\infty}/2 \leq \widetilde{V} \leq \widetilde{V}_{\infty}$, when the perturbation of the electrical pulse decreases the vortex radius. On the contrary, when the electrical pulse decreases the vortex radius, we select values of $\widetilde{V}$ such as $\widetilde{V}_{\infty} \leq \widetilde{V} \leq 2\widetilde{V}_{\infty}$. These choices can be summarized by the following criteria which is applied in any case to select the point for the fit:

$$\left| log\left(\frac{\widetilde{V}}{\widetilde{V}_{\infty}}\right) \right| \leq log(2) \tag{75}$$

Figure 75 is a zoom on the relaxation phase of Figure 74. The points used for the exponential fit are highlighted in green. The exponential fit is plotted in red. The value of T_{relax} is extracted from this fit. The portion of the curve in blue corresponds to values of $\widehat{V}$ which do not respect the criteria of equation 75. Therefore, they are ignored for the fit. In the criteria from equation 75, we choose for the fit values of $\widetilde{V}$ that are at the most twice larger or twice smaller than $\widetilde{V}_{\infty}$ to ensure that variations of u are small. Choosing values larger than 2 in this criteria, in order to reduce the variations of u, decreases the number of points used for the fit but does not significantly change the value of the extracted relaxation time. This measurement was repeated for different operating points and the results will be presented in the next section.

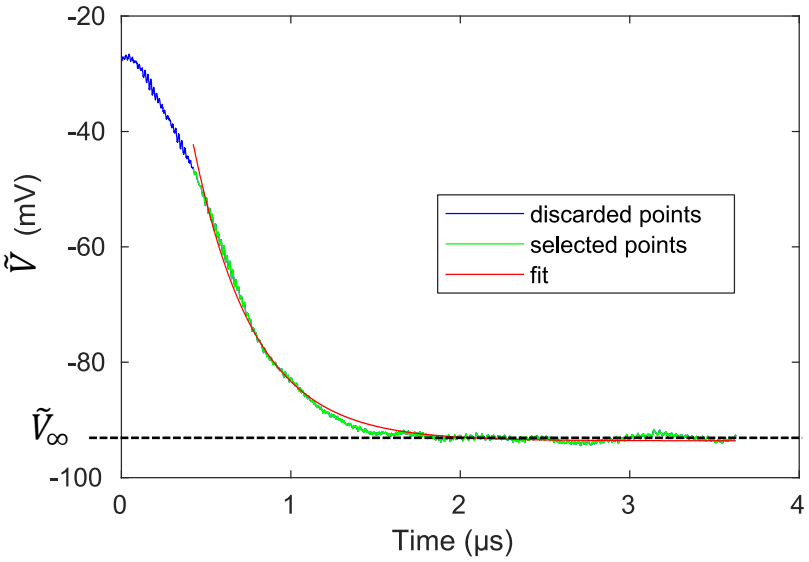

Figure 75: Zoom on the relaxation phase of $\widetilde{V}$, for the operating point H=300 mT and I=-5.5 mA. The portion of the time trace $\widetilde{V}(t)$ used for evaluation of the relaxation time is highlighted in green. The exponential fit is plotted in red. The portion of the curve in blue is ignored for the fit because it differs too much from the steady state value $\widetilde{V}_\infty$.

10.1.2 *Operating point dependent relaxation time*

T_{relax} is measured for d.c. currents varied from 3.5 mA to 9.0 mA and magnetic fields varied from 200 mT to 600 mT. Figures 76a and 76b show these results for, respectively, input spikes favoring and against the damping. All the white areas on the maps correspond to operating points where the relaxation time could not be evaluated because there was no emission, the signal over noise ratio was too low or the behavior of $\widetilde{V}$ could not be fitted by an exponential. The way the error on the fit propagates on the evaluation of the relaxation time is difficult to evaluate, because it requires numerous measurements (repeating the measurement many time and analyzing the variation of the measured noise on a significant amount of measurements) or extensive computations (in order to generate artificially "repetition of the same measurements" bootstrap algorithm [270] for instance can be used). We did not perform such systematic measurements and

thus we do not give a precise error bar on the measured relaxation time. But in order to have relaxation time that can be trusted, we only conserved relaxation time from measurements where the error on the fit ϵ_{fit} is inferior to 1.5×10^{-3}. We define ϵ_{fit} as:

$$\epsilon_{fit} = \frac{1}{N}\left(\frac{\|Y_{fitted} - Y_{fit}\|}{\max(Y_{fit}) - \min(Y_{fit})}\right) \tag{76}$$

where Y_{fitted} is the vector of the measured data (here it is the measured $\widetilde{V}$), Y_{fit} is the vector of the values from the fit and N is the length of Y_{fitted}. The measured relaxation time typically evolves between 100

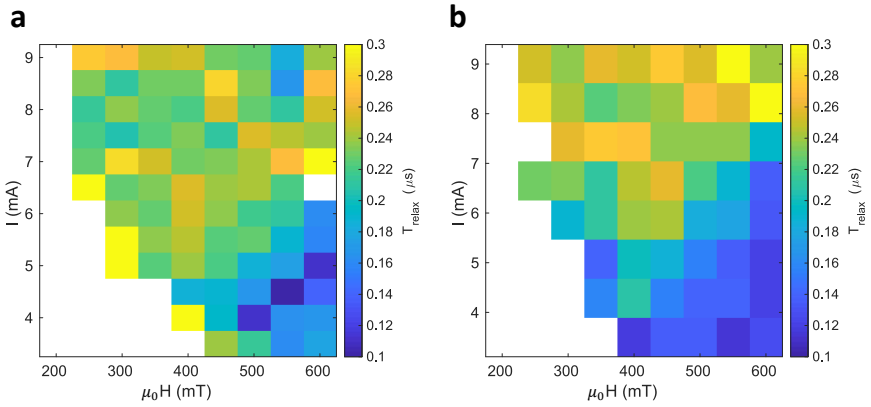

Figure 76: Relaxation time as a function of the operating point obtained for a. input spikes favoring the damping (positive voltage) and b. input spikes against the damping direction (negative voltage). White areas in the maps correspond to operating points where the relaxation time could not be evaluated because there was no emission, the signal over noise ratio was too low or the behavior of $\widetilde{V}$ could not be fitted by an exponential.

ns and 300 ns (Figure 76).

Even though the measured T_{relax} is similar for spikes favoring the damping (Figure 76a) or against the damping (Figure 76b), in some operating points these relaxation times differ, notably at low field (lower than 400 mT) and close to the critical current. For such case T_{relax} measured for spikes against the damping is much shorter.

Surprisingly, T_{relax} does not monotonically decrease with current as expected from theory [157, 153]. On the contrary we can observe that it increases again after 6.0 mA (Figure 77 in the case of H=600 mT),

which does not fit the expected $1/I_{DC}$ dependency predicted by theory (see for the theoretical expression of T_{relax} equation 33 in the uniform case and equation 50 in the case of a vortex in chapter 4). For 600 mT (Figure 77), the relaxation time increases drastically at 7 mA for spikes favoring the damping (red curve). This sharp transition could be due to a change in the magnetization mode. However this explanation remains an assumption. At 7 mA, the difference of measured relaxation time is particularly high ($T_{relax} \approx 250$ ns for spike against the damping and $T_{relax} \approx 580$ ns for spike with the damping).

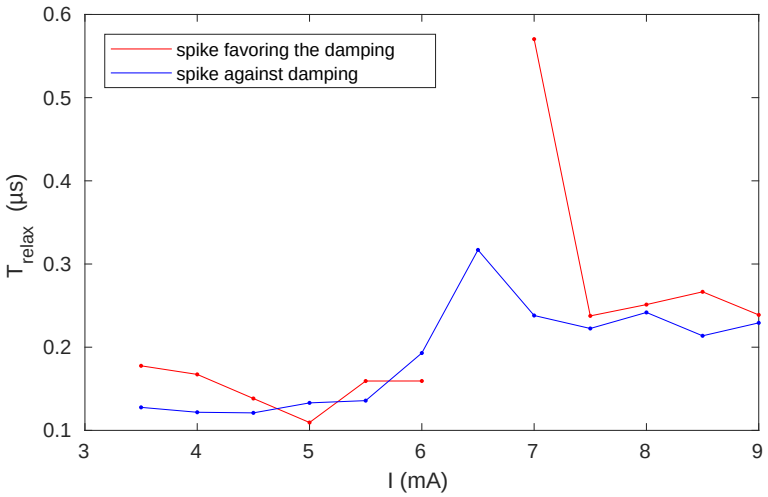

Figure 77: Relaxation time as a function of the I for H=600 mT for input spikes favoring the damping (red curve) and against the damping (blue curve).

In conclusion, the relaxation time of the oscillator was measured and varies typically between 100 ns and 300 ns. The dependence of the relaxation time on the operating point presents a quite unexpected behavior, since it does not decrease monotonically with current. T_{relax} was evaluated by sending spikes favoring and against the damping. It resulted in significantly different values of T_{relax} in some operating point conditions (notably close to the critical current), that may suggest an asymmetry in relaxation dynamical behavior depending if the vortex core goes to the center of the dot or in the opposite direction. These results are still preliminary and would require further studies, to be better understood. Nevertheless, the range of the intrinsic memory

of the oscillator, which corresponds to the typical relaxation time, was quantified to be around 200 ns. It corresponds approximatively to the expected order of magnitude of the relaxation time. Indeed in order of magnitude $T_{relax} \approx \frac{1}{\alpha\omega_0}$ (see equation 33). With $\alpha \approx 0.01$ and $\omega_0 \approx$ 300 MHz, a relaxation time around 300 ns was expected.

10.2 Implementation of extrinsic memory through delayed feedback loop

10.2.1 *Experimental set up*

In this section, we present the experimental circuit used for adding memory to the reservoir through a delayed feedback (Figure 78). The main difference with the circuit presented in chapter 6 (Figure 42) is the addition of the delay loop, composed of an electronic delay line with a delay $\tau = 4.3$ μs and a total amplification of about 20 dB. The addition of the delayed signal to the input is made with a power splitter. The reinjected signal is proportional to $\widetilde{V}(t-\tau)$ (the signal emitted by the oscillator and delayed of τ) and therefore, its level depends on the operating point. For the highest emitted voltage (measured at low magnetic field), the injected signal amplitude can reach approximately 50% of the amplitude of the input signal (250 mV peak to peak). Finally, the oscillation amplitude is recorded with an oscilloscope.

10.2.2 *Fading response to a single spike*

In this section we show the influence of the delay loop on the oscillation amplitude $\widetilde{V}(t)$ when a single spike is injected to the oscillator. The intrinsic memory of the oscillator and the feedback memory are illustrated in Figure 79. A 200 ns long current pulse is sent to the oscillator (Figure 79a) and the voltage amplitude of the oscillator $\widetilde{V}(t)$ is recorded by an oscilloscope (Figure 79b) during a much longer time (20 μs), in order to observe the reinjection effects and the memory induced by the delay line. Similarly to section 10.1, the input spike modifies the orbit of the gyrotropic motion of the vortex core. This changes the amplitude of the signal emitted by the oscillator $\widetilde{V}$. After

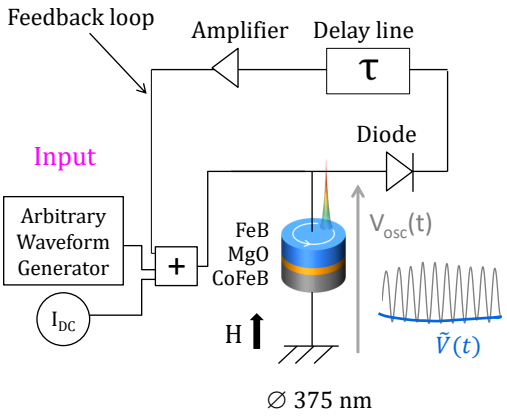

Figure 78: Experimental set up: The spin-torque oscillator receives a d.c. current and a perpendicular magnetic field which set the operating point. It emits an oscillating voltage $V_{osc}(t)$. The time-varying input is generated by an arbitrary waveform generator. A diode allows measuring directly the amplitude of oscillations $\widetilde{V}(t)$, which is used for computation. The feedback loop consists of an electronic delay line ($\tau = 4.3$ μs) and an amplifier (the total amplification in the line is +20 dB). The signals are added with power splitters.

the spike, the vortex core returns to the orbit defined by the fixed magnetic field and the d.c. current: the amplitude of the oscillations $\widetilde{V}(t)$ returns progressively to its initial level. During this time, which corresponds to the relaxation, the oscillator still remembers the input because the oscillation amplitude has not returned to its initial level. As seen in 10.1, the relaxation time of magnetization dynamics is around 200 ns, except for low fields and low currents or for high field (600 mT) after 6 mA. This intrinsic memory is highlighted in blue in Figure 79b. The feedback is implemented by propagating the perturbation of the oscillation amplitude in the delay line for $\tau = 4.3$ μs and reinjecting it into the oscillator. This injection induces new variations in the amplitude of the oscillator response $\widetilde{V}(t)$. Indeed, echoes in the oscillator response are observed every τ after the end of the input signal. These echoes are the manifestation of the external memory provided by the delayed feedback. Since after each τ the echo is more and more attenuated, the feedback memory is a fading

memory. The fact that memory fades is important to process temporal sequence for which only the recent history is important to ensure good classification performance [239]. For a magnetic field of 600 mT and a d.c. current of -6.5 mA, we observe variations in the oscillator output even after 13 μs (approximately 60 relaxation times). The range of the feedback memory depends on the delay in the line. Here we observe a memory of 13 μs, which corresponds to three times the delay of the line, but choosing a longer delay in the line would have resulted in a longer range of memory. Moreover, the feedback memory is sparse, that is, the information that the oscillator received as input in the past is only accessible every time τ (4.3 μs in our case) in the areas highlighted in orange in Figure 79b. In between, this information cannot be extracted from the measured trace. If the input is discrete, the time step of the input and the delay in the line are often chosen to be equal, for the oscillator to remember the input at previous time steps.

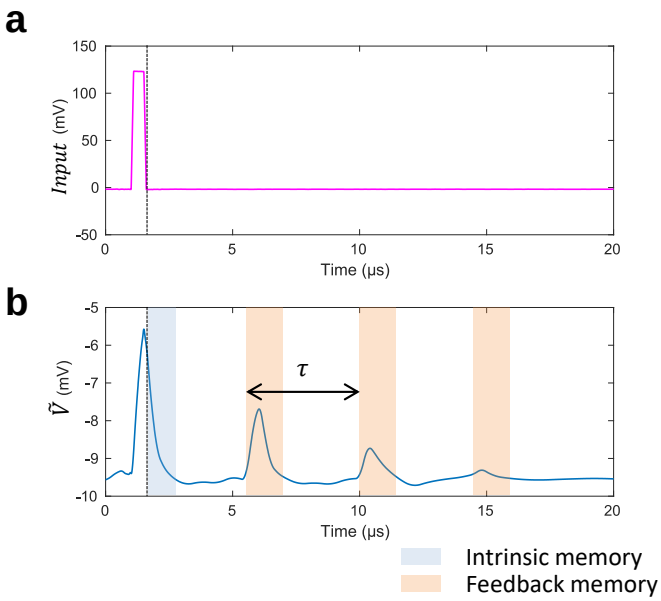

Figure 79: (a) Input spike (in magenta) sent by the arbitrary waveform generator to the spin-torque oscillator. (b) Blue curve: variation in the amplitude of the delayed feedback oscillator response $\widetilde{V}(t)$. The shaded areas in blue and orange indicate respectively the intrinsic memory and the feedback memory. The operating point is 600 mT and -6.5 mA.

In the particular case of Figure 79 three additional peaks were observed, but depending on the operating point, the number of observable peaks varies. Figure 80 shows the variation in the amplitude of

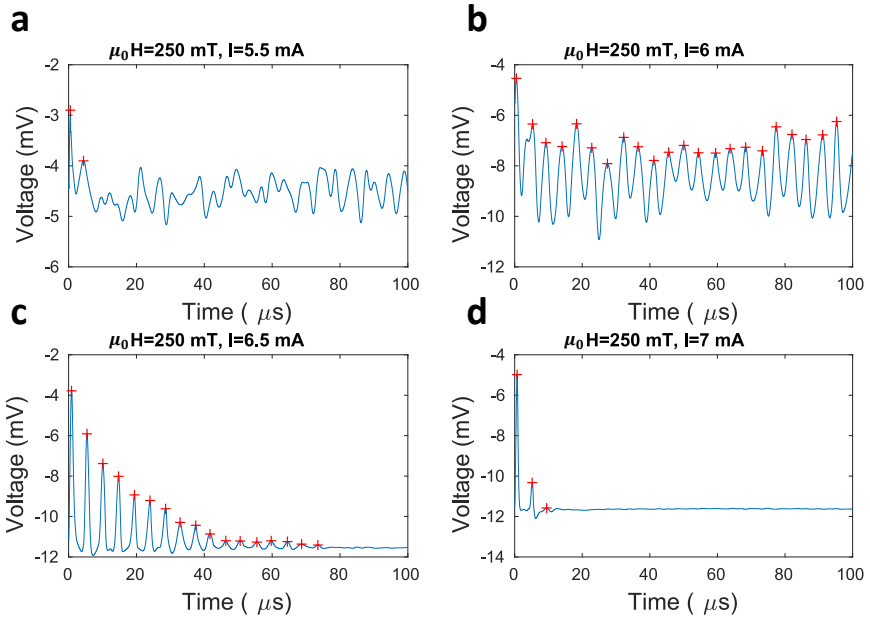

Figure 80: Variation in the amplitude of the delayed feedback oscillator response $\widetilde{V}(t)$ for a field of 250 mT and various d.c. currents: a, 5.5 mA b,6.0 c, 6.5 mA d, 7.0 mA.

the delayed feedback oscillator response $\widetilde{V}(t)$ for a field of 250 mT and various d.c. currents. For the lowest current -5.5 mA (Figure 80a), the oscillator is in a regime just above the oscillation threshold where it is very sensitive to perturbations. Thus the variations in $\widetilde{V}(t)$ are not correlated with the reinjection. By increasing the current to -6.0 mA (Figure 80b), $\widetilde{V}(t)$ oscillates with maxima every time τ (highlighted by a red cross), but the amplitude of these oscillations is not related to the distance to the first initial peak. In this situation, it is not possible to know when the delayed feedback oscillator received a perturbation, just by observing one maximum amplitude. Therefore, if multiple perturbations happen in the past, they can not be deduced from the shape of the present $\widetilde{V}$ and even though multiple spikes are observed, they cannot be used for tasks requiring memory such as the memory task developped in the next section. This case illustrates the necessity

for the memory to fade. For -6.5 mA (Figure 80c) a fading response is observed with a large number of visible peaks (16 echoes of the initial variation). Finally, increasing the d.c. current to -7 mA decreases the sensitivity of the oscillator to perturbations and only two echoes are observed. The number of observed echoes depending on the operating point is summarized in the following map (Figure 81).

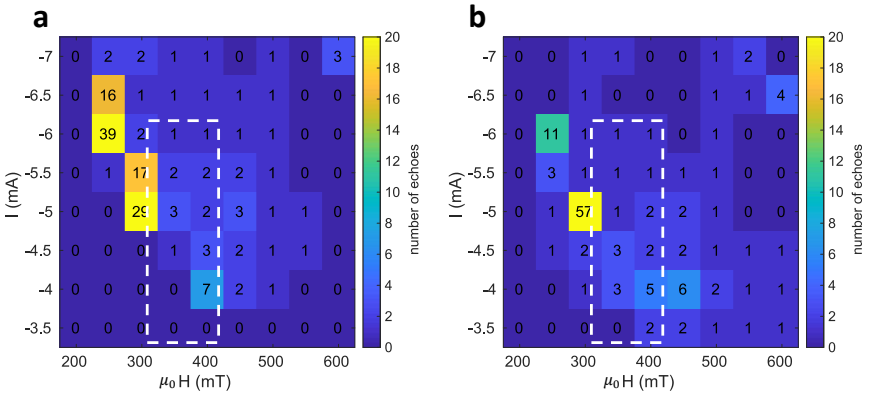

Figure 81: Number of echoes in the amplitude variation $\widetilde{V}(t)$ after a perturbation favoring the damping (a) and against the damping (b) as a function of the operating point. The white dashed line rectangle correspond to the range of operating point were the memory capacity was computed.

The number of observed echoes is evaluated in the case of a perturbation in the damping direction (Figure 81a) and against the damping (Figure 81b). The largest numbers of echoes are observed close to the critical current, since in this region, the oscillator is more sensitive to the external fluctuations. For a same operating point, the number of observed echoes depends of the sign of the perturbation, because the part of the $\widetilde{V} = f(I)$ function which is explored is not the same, resulting in different amplitude variations. This observation is similar to the asymmetry observed between V_{up} and V_{dw} in chapter 9 (see for example Figure 64).

In conclusion, a delay loop brings an additional memory, because echoes in the oscillation amplitude are observed after the end of the perturbation. The periodicity of the echoes is the value of the delay in the loop and the number of observed echoes gives an intuition of the range of this memory. However it is not clear if the initial pertur-

bation can be retrieved from the echoes. For instance, in Figure 80b, the amplitude of the echoes does not provide an information on the amplitude of the input.

10.3 Evaluation of the memory capacity of an oscillator

In this section, we compute the memory capacity of a delayed feedback spin-torque oscillator. It evaluates if a past input value can be retrieved from the present response of a dynamical system.

10.3.1 *Task*

This task was introduced by Jaeger [240] to evaluate the memory of a reservoir network. In this subsection, therefore, the delayed feedback oscillator will be used with time multiplexing to perform reservoir computing on the memory task.
The input stream $u(k)$ to be processed is a random sequence of values +1 and -1 (see Figure 82a). The goal of the task is to retrieve from the reservoir state $X(k)$ the value of $u(k-l)$, where l represents how far in the past is the input value to retrieve. We call y_l the output reconstructed from $X(k)$. y_l is the same sequence as u but shifted of l time steps to the right. For $l = 2$, the output that should be reconstructed is illustrated in Figure 82b.

The rest of the process is very similar to any other task of reservoir computing. We determined $W_{res}^{out}(l)$ and $W_{bias}(l)$ using training examples for each l value. They should minimize the error $\|W_{res}^{out}X(k) + W_{bias} - u(k-l)\|_2$. We define $y_l(k) = W_{res}^{out}(l)X(k) + W_{bias}(l)$. The memory performance $m(l)$ on n_{test} examples to retrieve the input l steps in the past is computed as the cross correlation between the $y_l(k)$ values and the $u(k-l)$ values:

$$m(l) = \frac{\sum_1^{n_{test}}(y_l(k) - \overline{y_l})(u(k-l) - \overline{u})}{\sqrt{\sum_1^{n_{test}}(y_l(k) - \overline{y_l})^2 \sum_1^{n_{test}}(u(k-l) - \overline{u})^2}} \tag{77}$$

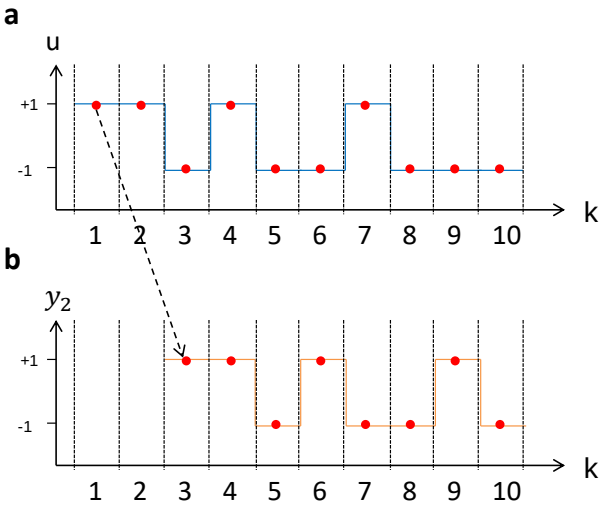

Figure 82: (a) Input $u(k)$ (b) output y_l in the case of $l = 2$.

10.3.2 *Results*

We performed this task using a delayed feedback spin torque oscillator as a reservoir. By comparison with the previous section, the delay used during this experiment is shorter (1.7 μs). The reason for this different delay time is only due to the planning of this work. The experience for memory task was performed before the study of section 10.2. At that time, we did not have all the delay lines necessary to build a longer delay of 4.3 μs. 24 temporal neurons are used for this task. We use the constant $\tau = 1.7$ μs for the preprocessed input (the time scale of the input is equal to the delay in the feedback loop) and $\theta = 72$ ns ($\tau/24$). The input is composed of 100 values, and half is used for training and the other half for testing. This input is preprocessed using 24 temporal neurons with the same mask used for the sine/square classification. The memory capacity $m(l)$ was computed for l varied between 0 and 9. The memory was tested for 3 values of the magnetic field 330 mT, 380 mT and 430 mT and for d.c currents from 2.8 mA to 6.0 mA (see white dashed line rectangles in Figure 81). The highest $m(l)$ values were obtained for a field of 330 mT and a d.c. current of 5.4 mA which is consistent with the map obtained in Figure 81. Indeed, in a neighboring region at 300 mT and 5.5 mA, 16 echoes were counted in previous section 10.2. So in this region of the

H-I_{DC} plan, the oscillator has an optimal sensitivity to perturbations for feedback memory. These memory capacity results are plotted in Figure 83. As it can be observed, the correlation between the output

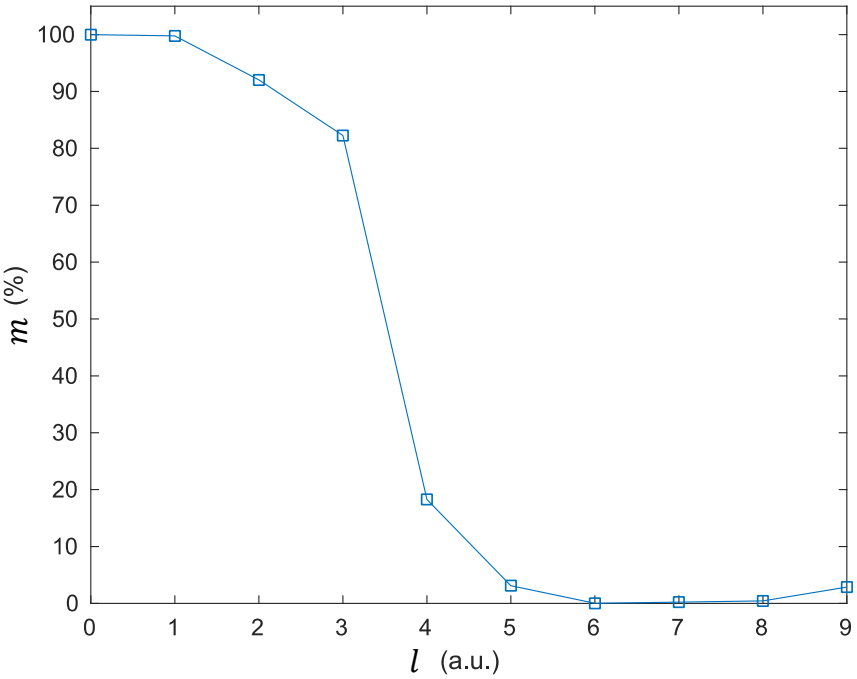

Figure 83: Experimental memory capacity m as a function of l. For this measurement the operating point is 330 mT and 5.4 mA. The delay of the delay line is 1.7 μs and 24 temporal neurons are used.

$y_l(k)$ and the past input $u(k-l)$ is high for $l \leq 3$. The information from the delayed feedback oscillator allows to retrieve past inputs up to three time steps in the past, but previous input values cannot be retrieved. For the time constant chosen in this study, this means it can retrieve an input value 5.2 μs in the past. This time is much larger than the relaxation time ($\approx$ 200 ns), so this memory effect is entirely due to the feedback. We obtain, here, a smaller memory than the maximum number of observed echoes. Also, this evaluation of memory capacity was not performed for as many operating points as the one presented in section 10.2. Moreover, this method to evaluate the memory of the delayed feedback oscillator is more demanding than counting the echoes, as done in the previous section. These two points can explain the smaller range of the evaluated memory with

memory capacity, when compared with previous section. Nevertheless, we demonstrate here that the delay feedback memory allows to properly retrieve multiple informations in the past.

10.4 CONCLUSION

In the first section we measured the relaxation time of the spin-torque oscillator, in order to evaluate the range of intrinsic memory. Relaxation time was measured between 100 ns and 300 ns, with an unexpected non-monotonic behavior, compared to theory, and with some asymmetry of its value, depending whether the vortex core relaxes towards a position closer or farther away from the center of the dot.
In the second section, we proposed to add a delayed feedback to the oscillator in order to build an extrinsic memory. When the oscillator receives an input, multiple echoes of this perturbation, due to the delayed feedback, are observed in the oscillation amplitude response. A delay of 4.3 μs was chosen, allowing to observe marks of the perturbation after a time much longer than the relaxation time (for example 69 μs for 300 mT and 5.5 mA). The number of observable echoes depends on the operating point, because the more the oscillator is sensitive to perturbations, the more echoes will be observed.
Finally we evaluate the memory capacity of a delayed feedback oscillator with a delay of 1.7 μs, in order to compute how well a past input can be retrieved from the delayed feedback oscillator response. This memory capacity is evaluated for input values that are older and older. The range of operating points tested was not as large as for the previous study. Nevertheless, we could evaluate a memory capacity, which is approximatively 3. The delayed feedback oscillator can retrieve input values at least three time steps in the past.
Also it is important to note, that the memory capacity demonstrates if a system can retrieve inputs, but it does not show if a system can perform recognition using past and present inputs. Indeed, the systems that perform the best at the memory capacity task are linear systems [240] that are unable to recognize complex patterns. The efficiency of this memory for recognition tasks remains yet to be demonstrated.

11

IMPROVED SEPARATION OF DATA IN CASES REQUIRING MEMORY

In this part the feedback effect as a memory is tested on a sine/square classification task. Firstly, we explain the interest of this task to test the memory. Secondly, we show the effect of feedback on time traces. Finally, we show the effect of feedback on the higher dimension reservoir state. The reservoir state will be visualized in two dimensions using projection methods. An improvement in the data separation will be observed due to this projection. The experimental data are obtained for the bias conditions H=300 mT and I=-6.5 mA.

11.1 EXPERIMENTAL PARAMETERS

In the previous chapter, the memory capacity of a spin-torque oscillator was studied. The memory capacity evaluates how far in the past the system can remember an input, but it does not show if this memory is usable for computation. Indeed, systems that perform the best at memory capacity task are linear systems [240], that cannot separate inputs. More generally, it has been shown that there is a trade-off between memory and non-linearity [271, 272, 273]. To test the memory in the context of pattern recognition, we performed the sine/square classification task, which requires both memory and non-linearity.

11.1.1 *Task*

The sine/square classification task was presented in section 7.1.1 (Figure 46). As mentioned in section 7.1.1 this task requires both non-linearity and memory. The input sequence to classify is made of the same succession of discrete periods of sines and squares randomly arranged as in chapter 7. The sine and square inputs take values between -1 and 1. The different input cases are referred as si1-8 for

the sine pattern and sq1-8 for the square pattern. These patterns present degeneracies because different input cases take the same value. In particular si3 and si7 take the values +1 and -1 just as respectively sq1-4 and sq5-8. To classify correctly si3 and si7, the reservoir should remember the previous input si2 and si4. In this situation the information to store in the reservoir is not one value, but a sequence of two consecutive values. For this task, it is not necessary to remember more than one step in the past.

11.1.2 *Preprocessing time constant*

The sine square classification task is treated similarly as in section 7.1.2. We use a single oscillator with a delayed feedback loop which emulates a 24 temporal neurons reservoir. As in chapter 7, the connection between the input and the reservoir is emulated by the preprocessed input $J(t)$. We choose to synchronize the delay feedback loop with the input. The feedback delay is of 4.3 μs, so we generate a preprocessed input $J(t)$ with a long time scale $\tau = 4.3$ μs. In this case, at time $t = (k-1)\tau + i\theta$, when the oscillator temporal state represents $x_i(k)$ (the neuron response i to the input $u(k)$), it is fed with $J((k-1)\tau + i\theta) = w_i^{in} u(k)$ and $x((k-2)\tau + i\theta) = x_i(k-1)$. So, in the end, the temporal neurons feedback themselves and remember their own previous response. The equivalent spatial topology is represented in Figure 84, where, for simplicity, a reservoir with only five temporal neurons (blue circles) was represented. The connections between temporal neurons are represented by dashed blue arrows in Figure 84, and are ensured by the transient dynamics of the oscillator (with the time scale θ). The connections brought by the τ long delay line are represented by solid line feedback arrows on the temporal neurons. Then the output is computed off-line (similarly to the method presented in chapter 7) by linear combination (red arrows). In the present study, the time θ is 180 ns ($\tau/24$), which is larger than the one chosen in the previous chapters. It approaches the value of the relaxation time ($T_{relax} \approx 200$ ns), allowing to decrease the effect of the intrinsic memory of the oscillator. Indeed, in chapter 7, the oscillator could solve the sine/square classification task leveraging only the intrinsic memory of the oscillator. In the present work, our goal is to study the effect of feedback as a usable memory for computation. The

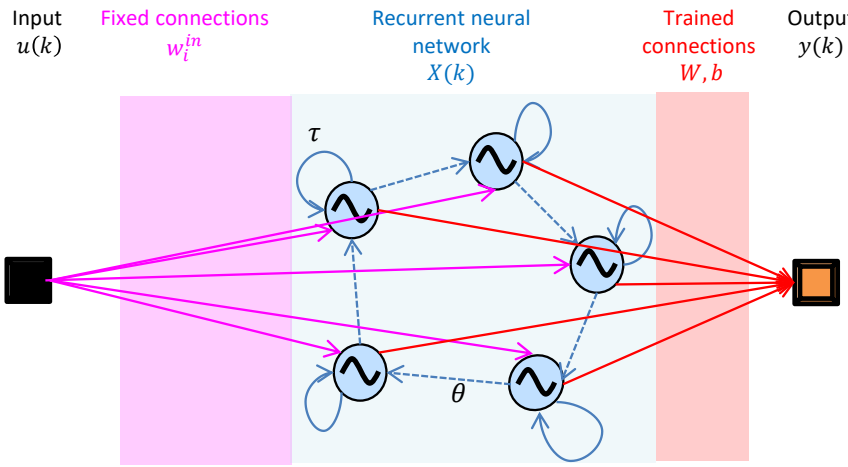

Figure 84: Equivalent spatial reservoir: blue circles represent neurons, dashed line arrows are the connections between consecutive neurons and solid line arrows are feedbacks for the neurons.

intrinsic memory effect is here voluntarily attenuated so it does not screen the effect of the feedback memory.

11.2 TIME TRACES

In this section, we present the experimental time traces obtained for the different input cases. The operating point is, for this study, $\mu_0 H = 300$ mT and $I = -6.5$ mA, where the feedback effect has a particularly positive effect on classification (for more details, see chapter 12). A particular focus is made on the si3 and si7 input cases, as they are the two cases requiring memory to be distinguished from the square cases. Figure 85 shows the average time traces (colored solid lines) and twice the standard deviation (shaded region) for the oscillator amplitude response $\widetilde{V}$ when the input is the si7 case (Figure 46), compared with the sq5-8 cases (blue-green color). This curve is obtained by averaging all the time traces when the input is a si7 (80 cases) and computing the standard deviation over all these traces at each time step. For this first case (without feedback), we can see that the trace for si7 is overlapping with sq6-8 (the overlap is particularly important for sq7 and sq8). The sq5 case differs slightly

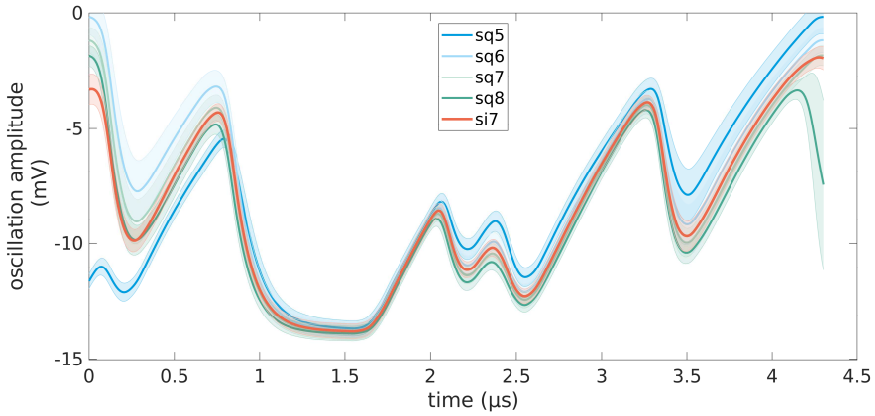

Figure 85: Time traces without feedback.

at the beginning of the time trace where small effects of the relaxation are still present (we chose $\theta \approx T_{relax}$ and not $\theta >> T_{relax}$ for practical reasons). For sq5, the previous value is -1 (sq4) so the transition is more important. This difference disappears quickly after the first θ. Figure 86 shows the same average time traces, in the case of a delayed feedback. The time traces differ more from each other. In particular, the si7 time trace distances from the sq5-8 time traces. Among the sq time traces, sq5 becomes very different from sq6-8. Importantly, the time traces do not need to be different at all times in order to lead to separable reservoir states. Indeed the reservoir state is obtained by sampling these curves at different times.

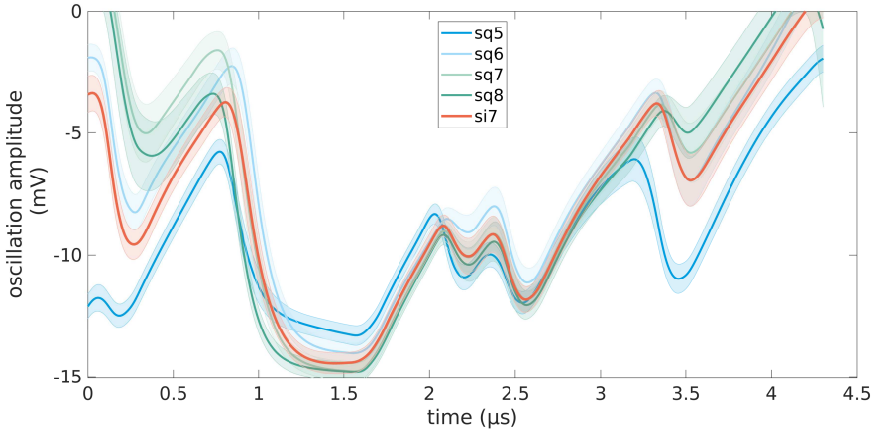

Figure 86: Time traces with feedback.

The case of si3 presents similar features. Figure 87 shows the si3 case (orange) time traces with respect to sq1-4 case without feedback (blue-green curves). The si3 case time trace completely overlaps with sq1-4 cases. Compared with the previous ones, the dispersion of the time traces (shaded area) is much lower because most of the time trace is in a region of high emission (which was not the case before). The oscillation amplitude $\widetilde{V}$ is here in the range -10,-15 mV. As seen in chapter 8, in this region the function $\widetilde{V} = f(I)$ varies slowly which means it is less sensitive to thermal fluctuations.

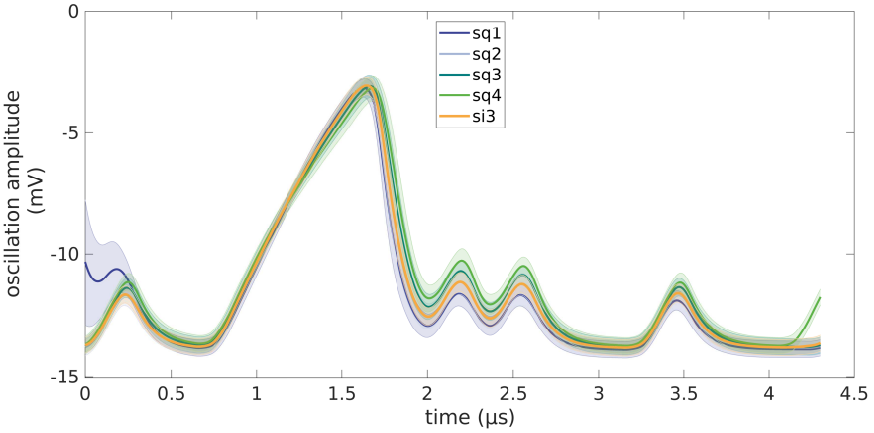

Figure 87: Time traces for si3 without feedback

When feedback is added to the oscillator (Figure 88), the si3 case differs from the sq cases. Also, the differences are less important than in Figure 86, once again because in most of the time traces $\widetilde{V} = f(I)$ varies slowly, so the reinjected signal does not induce large variation of the $\widetilde{V}$. The largest differences between the curves are around 1.5 μs where emission is lower ($\widetilde{V} \approx -5$ mV). The lower variation between the average time traces is counterbalanced by the lower dispersion of each input case time traces.

As seen in the previous chapters, these time traces represent in time the higher dimension mapping of a reservoir. In our case, each time trace represents a 24 dimensions reservoir state. In the next section, we will study how the memory modifies the distribution of the input cases in the reservoir state.

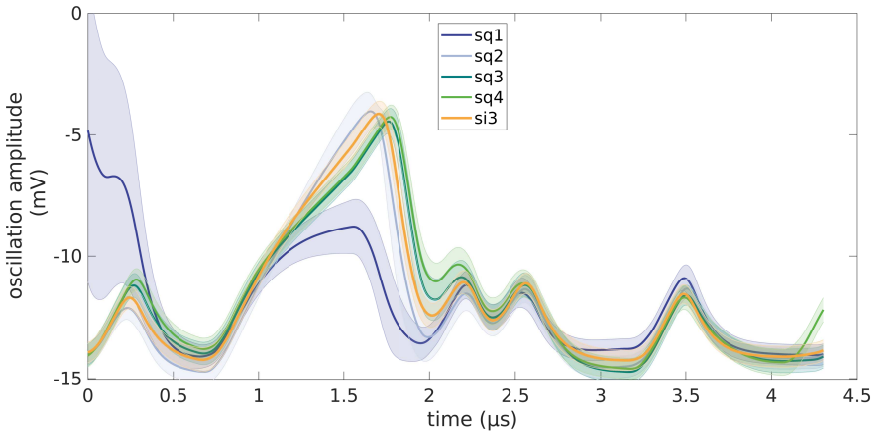

Figure 88: Time traces for si3 with feedback.

11.3 EXPECTED DISTRIBUTION OF THE INPUT CASES IN THE RESERVOIR STATE DEPENDING ON THE MEMORY

In this section we will detail which clusters are expected to form in the reservoir higher dimension mapping, in the case the reservoir possesses a memory or not.

In Figure 89, the pattern to recognize is plotted and the different input cases that should form a same cluster without memory are highlighted.

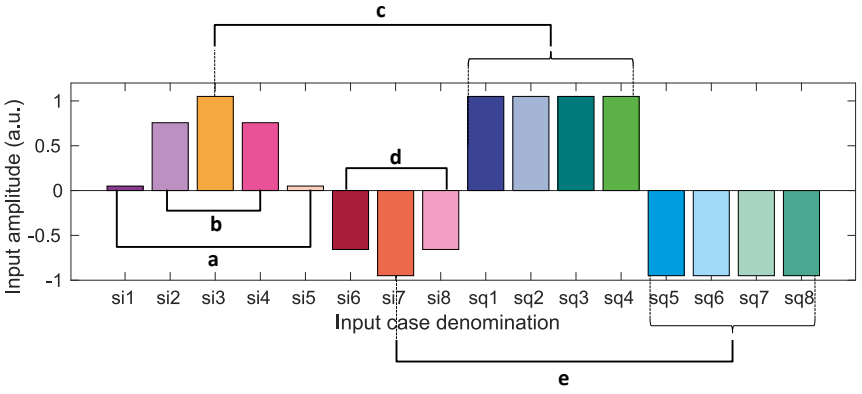

Figure 89: 2 dimension projection of reservoir mapping without memory: The input pattern takes five different values, so five groups of input cases (a-e) are expected.

The input pattern exhibits different input cases that are assigned to the same value, thus, it possesses degeneracies. The input pattern takes 5 different values (0, 0.7, 1, -0.7 and -1) so, with an ideal input, without noise in the reservoir behavior, and in the absence of memory each of the different values should induce a particular reservoir state. Of course because of the noise in the input and in the response of our spin-torque oscillator, the time trace for a same input case fluctuates (as observed in the previous section). So we do not expect to observe exactly a unique reservoir state but more point clusters whose widths correspond to the dispersion of the time traces. In the case of sine/square classification, the reservoir states cannot perfectly separate the square cases and the sine cases in the absence of memory because si3 (si7) induces the same reservoir state as sq1-4 (sq5-8). Indeed as observed in the previous section, in the absence of intrinsic memory for large thetas, the corresponding time traces completely overlap.

If the reservoir can remember one time step in the past, when the reservoir receives an input, its state is determined by the input current value and the previous one. The number of different clusters of possible reservoir states is then determined by the number of different sequences of two consecutive input values present in the input. For sine and square inputs (discretized in 8 points), 14 possible

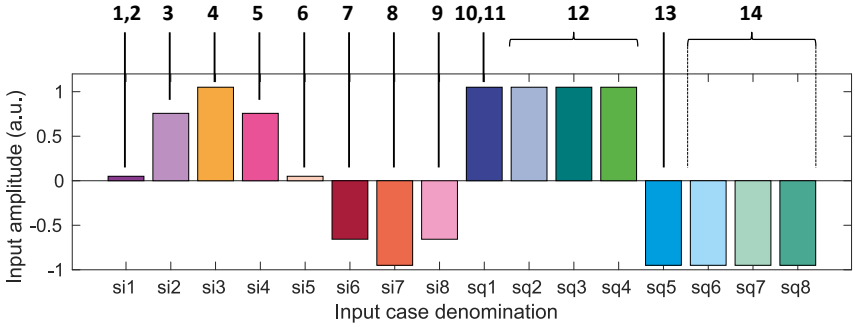

Figure 90: Expected mapping with one time step memory: the input pattern possesses 14 possible sequences of two consecutive values. So 14 groups of input cases (1-14) are expected.

different sequences appear in the input pattern. These sequences are denominated with numbers from 1 to 14 in Figure 90 and not by letters like in previous cases in order to not mix up the cases with

memory and without memory. In particular, si1 and sq1 cases give rise to different reservoir states 1,2 and 10,11. Indeed, for si1 depending if the previous input in the signal is a sine or a square we have two different sequences of values [-0.7,0] and [-1,0]. In a similar fashion, for sq1 the possible sequences are [-0.7,1], if the previous period is a sine, and [-1,1], if the previous period is a square. sq2-4 (sq6-8) should still give similar reservoir states because they correspond to the same sequence [1,1] ([-1,-1]).
If the reservoir remembers more time steps, the number of possible clusters increases. For instance, for a two time steps memory, sequences of three consecutive values should be considered and eighteen clusters should be expected.
The memory allows separating sine and squares because, in the case of a one time step memory, there is no overlap between the reservoir states induced by sine input cases and by square input cases. It also increases the number of different clusters of reservoir states induced by the input pattern. In this section, no assumptions on the reservoir were made besides its memory range. These predictions are thus general. In the following sections we will confront these predictions with our experimental results. In particular we will use a 2 dimension visualization to observe these expected clusters.

11.4 2 DIMENSION VISUALIZATION OF THE RESERVOIR MAPPING

In order to study changes in the reservoir state due to reinjection, the 24 reservoir states are projected in two dimensions so they can be visualized on a map. First, we explain the principle of the two dimensional projection we use. Second, we will show in two dimensions the reservoir state corresponding to the time traces we have seen in previous section, which sheds light on the critical role of feedback for separating the data, that are only discernible with memory.

11.4.1 *2D projection method*

Since one of the goals of the 2D projection is to visualize the linear separation between the mapping of sine and square input, we chose

linear projection methods (contrarily to the t-SNE method that we used in chapter 9). The objective is thus to find two vectors to project the data along and, to avoid distortion, we chose orthogonal vectors. In addition, to show the separation between sine and square data, the projection should conserve all the different clusters of points that are formed. This last point is important, for instance, when separation is not perfect, in order to identify which input cases from sine and square are mixed up together. Each of the time traces presented in the previous section will be represented by a point in the 2D projection. The projection consists in computing the coordinate of these points. The first step is to extract the temporal neuron outputs from the time traces, by sampling points every θ. Each time trace in response to $u(k)$ is represented by a 24-values vector, which represents a reservoir state $X(k)$. Then, the first coordinate ν_1 of the point is computed by the linear combination:

$$\nu_1(k) = W_{res}^{out} X(k) + W_{bias} \tag{78}$$

where W_{res}^{out} and W_{bias} are the weights determined during training (see chapter 5 and 7). The first coordinate, ν_1, shows the separation between sine and square. The limit between sine and squares on the map is visualized by the line of equation $\nu_1 = 0.5$. Geometrically, ν_1 can be seen as a projection of $X(k)$ along the vector W_{res}^{out}. The bias W_{bias} just ensures that along this projection, the linear separation occurs at 0.5 (it can be seen as a change of origin). To ensure the orthogonality of the projection base, the second projection vector should be orthogonal to W_{res}^{out}. We look for the second vector in the space H, orthogonal to W_{res}^{out}, and we now study $X_\perp(k)$, the component of the reservoir state in H.

$$X(k) = X_\perp(k) + (\nu_1(k) - W_{bias}) W_{res}^{out} \tag{79}$$

The goal of the second coordinate ν_2 is to reveal the clusters in the data distribution of the $X(k)$ vectors. For this purpose, we take as second coordinate, ν_2, the first principal analysis (PCA) score on $X_\perp(k)$. PCA is a standard method used to decrease the number of variables in a problem. Let's define $M = [X_\perp(k)]_{k \in [1, N_{tau}]}$, the matrix

containing all the $X_{\perp}(k)$. PCA finds a vector u in H such as the variance of $X_{\perp}(k)$ projected allong u

$$var(\Pi_u) = var(M.u) \tag{80}$$

is maximal. The second coordinate, ν_2, is thus given by the projection along u.

$$\nu_2(k) = X(k)u \tag{81}$$

It is important to note that, even if ν_2 is designed to account for the variability in the data of M, since it is obtained by a linear projection, we lose some information. In the end we try to design a linear projection that represents the separation and the different clusters in the reservoir. Going from 24 dimensions to 2 dimensions can induce a loss of information. In the following section, this linear projection is used on the experimental data and it will be shown how it allows to visualize the reservoir state.

11.4.2 *Emergence of new clusters due to reinjection*

The following maps, which will be described, are obtained with the projection method described in the previous subsection, applied to all the time traces used during the experiment (1280). We use, here, the data both for training and testing to have more representative points in the figure. The objective is to see the separation among the data, operated by the reservoir, before any form of training, so mixing training and testing data does not introduce a bias. In the following Figures 91 and 92 each time trace, in response to an input $u(k)$, is represented by a point of coordinates (ν_1, ν_2). These points are represented with the color associated with the input case of $u(k)$. For example, if $u(k)$ is a si1 case, the reservoir state it induces is represented by a dark purple point (the color associated to si1) on the 2D map. The separation between sine and square cases is represented by the vertical line $\nu_1 = 0.5$. If the reservoir separates well the different inputs $u(k)$, all the points associated to sine inputs (represented in warm color) should fall on the left of the vertical line and each point associated to square inputs (represented in cool color) should fall on the right of the vertical line. The first two dimensions map (Figure 91) represents the reservoir states without any feedback.

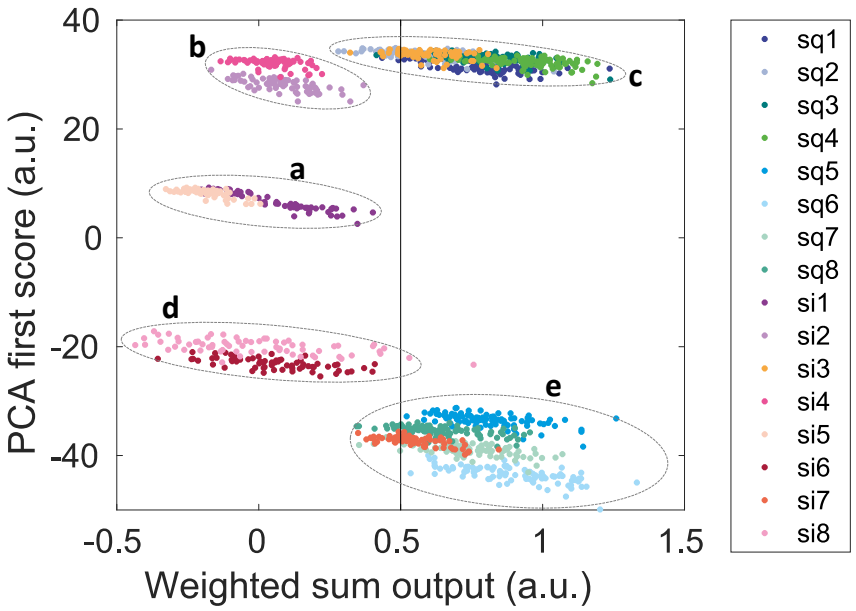

Figure 91: 2 dimension projection of reservoir mapping without feedback: gray dashed ellipses highlight the clusters corresponding to the different input value cases.

Figure 91 shows mainly five well separated clusters of data (highlighted with gray dashed ellipses) corresponding to the five different input values (cluster a-e) as predicted in section 11.1.1 (Figure 89). Also si4 and si8 seem to separate slightly due to the intrinsic memory effect. Similarly, the sq5 and sq6 cases seem quite dissimilar. But these differences remain small, when compared with the distances to other input cases. These first results show that, besides small differences observed in time traces, the effect of the intrinsic memory due to relaxation is mainly negligible. As expected, the si3 (si7) cases cannot be separated from the sq1-4 (sq5-8) cases. Two points of si8 cases are not well classified, because of a lack of reproducibility (noise). All the other 157 points that are not well separated come from a lack of memory. For these cases, training on 50% of data and testing on the remaining 50% lead to a 10.78% error rate on testing data (and a 12.66% error rate on training data).
Figure 92 represents the reservoir states when a feedback is added to the oscillator. The feedback notably increases the number of clusters in the data, showing that the feedback breaks the degeneracy of the

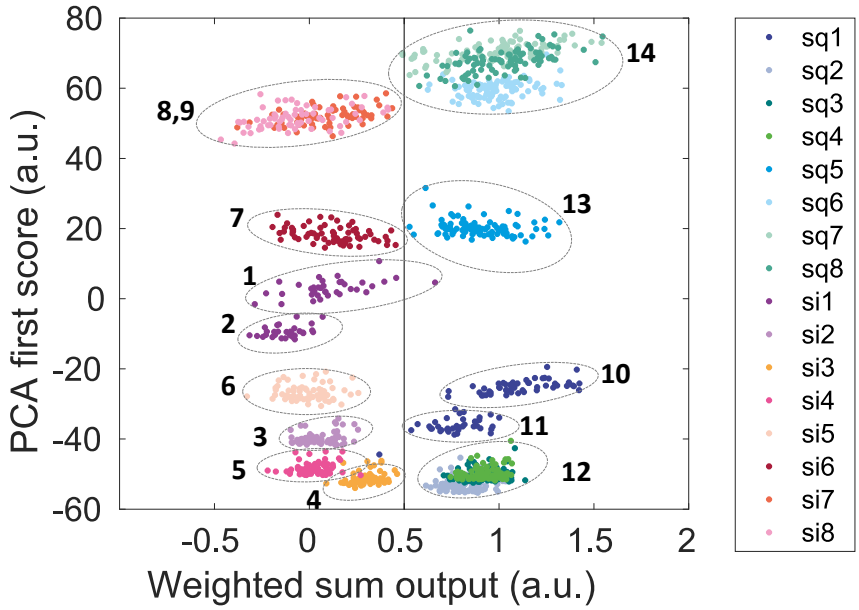

Figure 92: 2 dimensional projection of reservoir mapping with feedback: gray dashed ellipses highlight the clusters corresponding to the different input 2 values sequences.

input pattern. In particular, the si3 (si7) cases form a cluster clearly distinct from the sq1-4 (sq5-8) cases. Thirteen clusters are isolated in total. These clusters are consistent with a one-time-step memory. The thirteen clusters are associated with the cases seen in 11.1.1 (Figure 90). We associate the clusters observed experimentally with groups (1-14) that are expected with a one time step memory. All the expected groups are observed, besides groups 8 and 9 that are merged in a single cluster. This might be due to the loss of information that occurs during the projection. A linear separation can be drawn between sine and square cases with very few errors (4 in total). This higher dimension mapping of the data results in a high performance classification with a 0.14% error rate on testing (and 0.44 % errors on training). The feedback modifies the topology of the reservoir states because it provides memory. Feedback suppresses most of the degeneracies in the input pattern and creates new clusters among the reservoir states. Finally, a linear separation of sine and square inputs is achieved with a very low error rate.

11.5 CONCLUSION

Sine/square classification is an interesting task to evaluate the memory, because it requires both memory and non linearity. Moreover, the clusters of the reservoir states can be predicted, depending on the amount of memory.

Time constants τ and θ of the preprocessed input were specially designed to study only the memory brought by the feedback and not the intrinsic memory effects.

We have seen, in the second section, how the feedback modifies the time traces of $\widetilde{V}$. In particular, the time traces in response to si3 and si7 inputs (cases that require memory to be classified) become different from the square cases.

Finally, we designed a 2 dimension projection in order to visualize the 24 dimensions reservoir states on a map. The two coordinates of the map are constructed to visualize the separation between sine and square cases and, the different clusters of reservoir states.

These 2 dimensions maps confirm, without ambiguity, the memory effect of the feedback, because new clusters of data are formed in the reservoir state. In the end, the separation between sine and square goes from 10.8 % errors without feedback, to 0.14 % errors with feedback. These results prove that feedback memory can be used for computation.

The results presented in this chapter are obtained for an operating point of $H = 300$ mT and $I = -6.5$ mA. In the next chapter, we will see how the feedback memory is affected by the operating point.

12 INFLUENCE OF OSCILLATOR PROPERTIES ON CLASSIFICATION PERFORMANCE

In this chapter, we study the influence of the operating point on classification improvement brought by re-injection. In the first section, we present classification results with and without feedback and analyze which input cases these errors come. Then we link these errors with the operating point dependent properties of the oscillator and give general guidelines to choose well the regime of the oscillator when feedback is added.

12.1 CLASSIFICATION IMPROVEMENT

12.1.1 *Global classification result*

We have investigated the feedback effect in a large magnetic field range, from 200 mT to 600 mT, and DC current range, from -2 mA to -7 mA. Figures 93 and 94 show the error rate result depending on the operating point, respectively without feedback and with feedback. The situation without feedback (Figure 93) already presents disparities of result depending on the operating point. The red colors correspond to low error rates and the white color to high error rates. In particular, as seen previously, for low field and low current, high error rates are observed. For some operating points, such as 600 mT and -5.0 mA, on the contrary, low error rates are observed (5 %). These good results augur intrinsic memory effects even with the large value of theta that we use here ie θ =180 ns. These situations will be seen in more detail in the next section. When feedback is added (Figure 94) very low error rates are observed for many different operating points (dark color). Error rates inferior to 5 % (best error rate achieved without feedback) are highlighted in gray (15 operating point conditions).

Both the magnetic field and the DC current change the non-linear dependence of the voltage oscillation amplitude with the input current (see chapter 9), with the DC current acting as an offset for the input

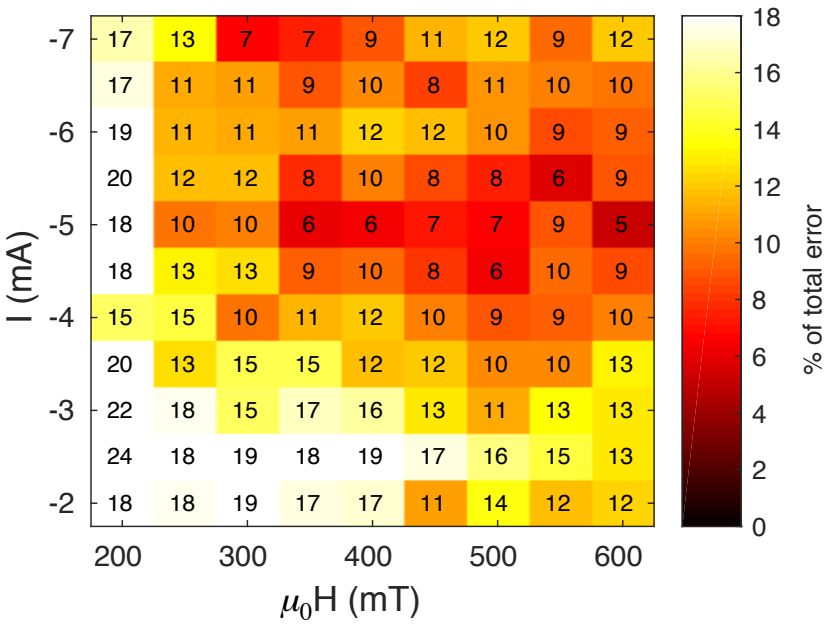

Figure 93: Error rate without feedback.

current. The improvement of the classification, which corresponds to the reduction of the error rate when feedback is added, depends on the operating point fixed by these two parameters (d.c current I and perpendicular magnetic field H). The normalized reduction of the error rate $\Delta\epsilon$ is computed as follows:

$$\Delta\epsilon = \frac{\epsilon_{NF} - \epsilon_F}{\epsilon_{NF}} \tag{82}$$

where ϵ_{NF} is the error rate in a situation without feedback and ϵ_{NF} is the error rate obtained in a situation with feedback. $\Delta\epsilon$ is thus positive when the feedback improves the classification and negative otherwise.

The error reduction when feedback is added, depending on the operating point, is summarized in Figure 95. Light colors correspond to situations where the feedback improves the classification and dark colors correspond to situations where the feedback degrades the classification. In particular white color corresponds to cases where the feedback suppress 100 % of the error and black color correspond to situations where the feedback increases the error by more than 100 %.

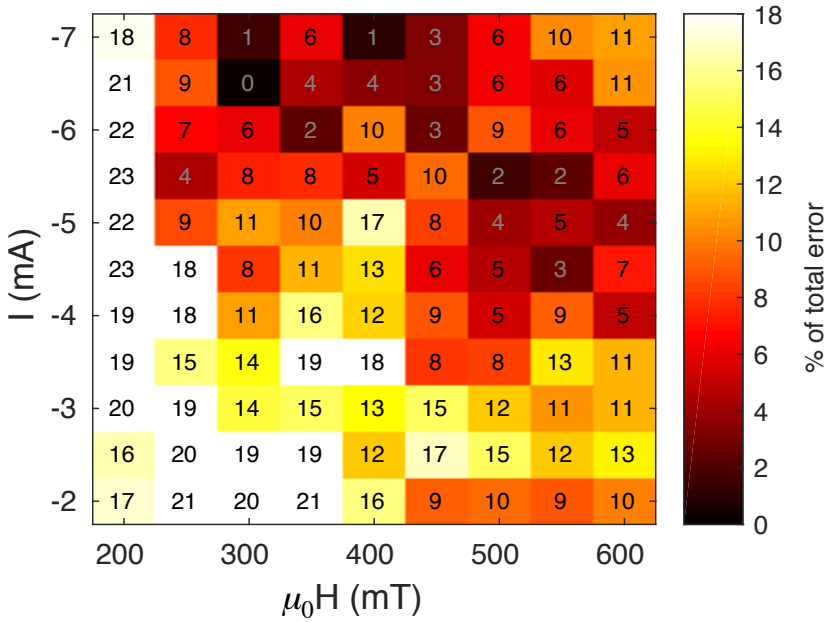

Figure 94: Error rate with feedback.

The highest improvement, detailed in chapter 11, is obtained for the operating point corresponding to 300 mT and -6.5 mA (Figure 95). In this case 99 % of the errors are suppressed by the feedback. As it has been seen this improvement is due to the suppression of the errors on si3 and si7, which are ambiguous without memory. More generally, as shown in Figure 95, the feedback reduces the error rate in 60% of the cases during the test phase ($\Delta\epsilon > 0$), it has no significant effect in 10% ($\Delta\epsilon = 0$) of the cases and it increases the error rate in 30 % of the cases ($\Delta\epsilon < 0$). In some cases, the feedback can have a very detrimental effect (for instance for 400 mT and -5.5 mA, the error increases by 159 % when feedback is added: $\Delta\epsilon = -159$ %).

The results in the testing phase depend on how the reservoir separates the data and how the separation learned during training is generalized to new examples. In order to simplify the analysis, we will focus on the separation part (as it is the heart of the reservoir computing method). So in the following error reduction maps, only the training data are used. Figure 96 shows $\Delta\epsilon_{train}$ as a function of

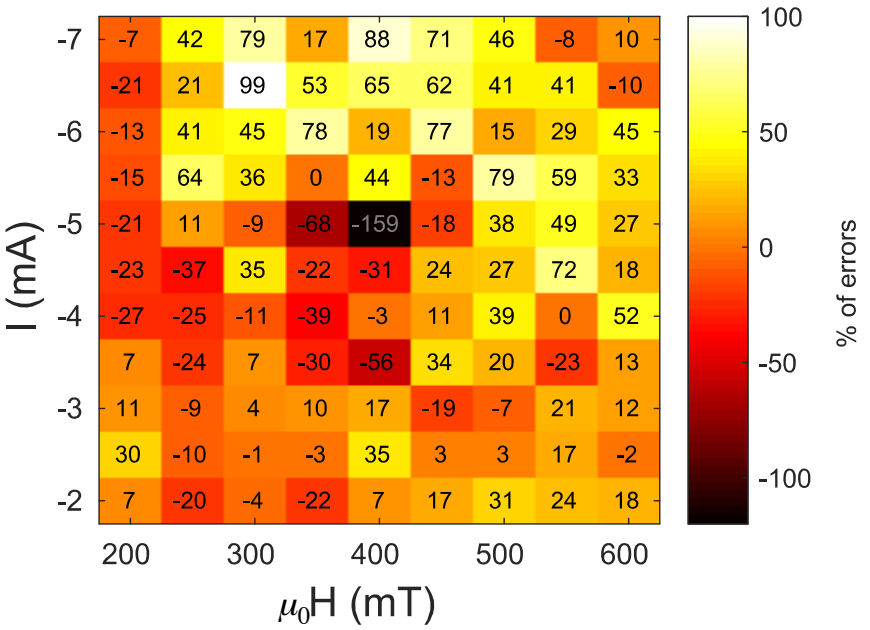

Figure 95: Normalized reduction of the error rate due to the feedback during the testing phase $\Delta\epsilon$ as a function of the operating point. Positive $\Delta\epsilon$ values correspond to a classification improvement due to the feedback.

the operating point. $\Delta\epsilon_{train}$ is the same ratio of error reduction as $\Delta\epsilon$ (see Figure 95) but using training data instead of testing data.

The variation of $\Delta\epsilon_{train}$ versus the operating point is plotted in Figure 96. It can be observed in Figures 95 that the variation of $\Delta\epsilon_{train}$ is similar to the variation of $\Delta\epsilon$ (Figure 96) and in particular for the training data the best improvement is also obtained for H=300 mT and I=6.5 mA. The final error rate is improved in only 66 % of the operating points during the training phase ($\Delta\epsilon_{train} > 0$), but when the feedback is detrimental during the training phase, the error rate degradation may be more pronounced during the testing phase: $\Delta\epsilon_{train} < 0$, $\Delta\epsilon < 0$ and $|\Delta\epsilon| > |\Delta\epsilon_{train}|$. For instance, at 400 mT and 5.5 mA, the feedback increases by 50 % the error during training ($\Delta\epsilon_{train} = -50$ %) and by 159 % during testing ($\Delta\epsilon = -159$ %). In this case, the feedback degrades the separation, since the error is higher during training ($\Delta\epsilon < 0$). The feedback also degrades the generalization because for the situation with feedback, the error rate

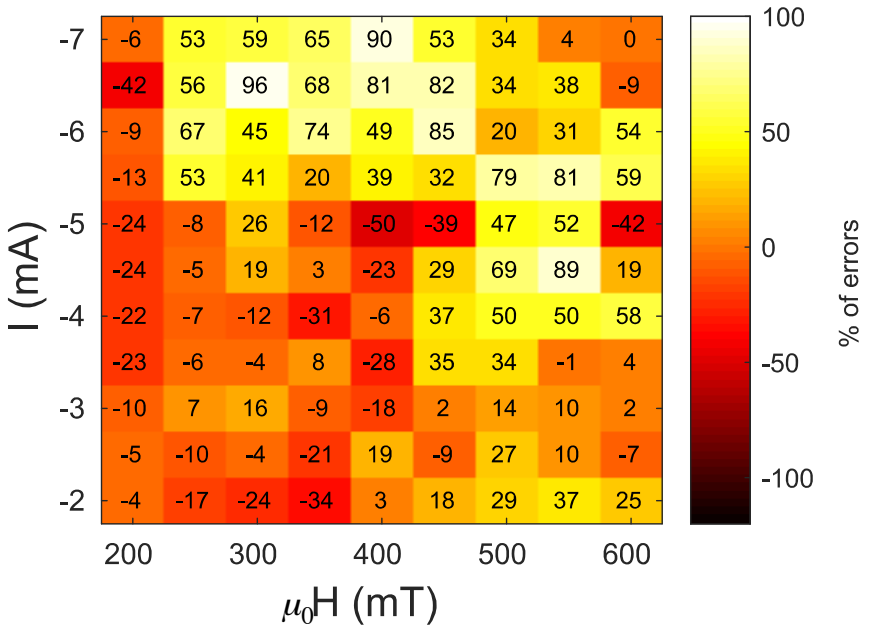

Figure 96: Normalized reduction of the error rate due to the feedback during the training phase $\Delta\epsilon_{train}$ as a function of the operating point.

during testing phase is much higher than the error rate during training phase, whereas without feedback the error rate during testing phase and training phase are similar. This last effect explains why $|\Delta\epsilon|$ is approximatively 3 times bigger than $|\Delta\epsilon_{train}|$. Therefore it is easier to analyze the situation of the training phase, because it focuses only on the separation and thus it suppresses the large negative values of $\Delta\epsilon$ which are due to a bad generalization and which are not easy to interpret.

In general, when the feedback improves the classification, it is related to the memory it brings because it allows classifying correctly the si3 and si7 input cases (see section 7.1.2). In the following section, we focus on the recognition of the si3 and si7 cases, in order to see if, depending on the operating point, the feedback assures its function as a memory for computation.

12.1.2 *Errors suppressed by the memory*

For this study we still focus on training data and we verify if si3 and si7 are well classified during training. If the reservoir states for the si3 and si7 cases are completely non discernible from those concerning the sq1-4 and sq5-8 cases, then the optimal weight W will classify the si3 and si7 as square inputs. Indeed, it generates 4 times less errors than classifying the sq1-4 and sq5-8 cases as sine, as such it is the optimal solution to reduce the global error. As a consequence, the number of misclassified si3 and si7 is a measure of the lack of memory. Here, we will compare the error on si3 and si7 without and with feedback (Figure 97). the feedback suppresses errors for si3 and si7 in 90 % of

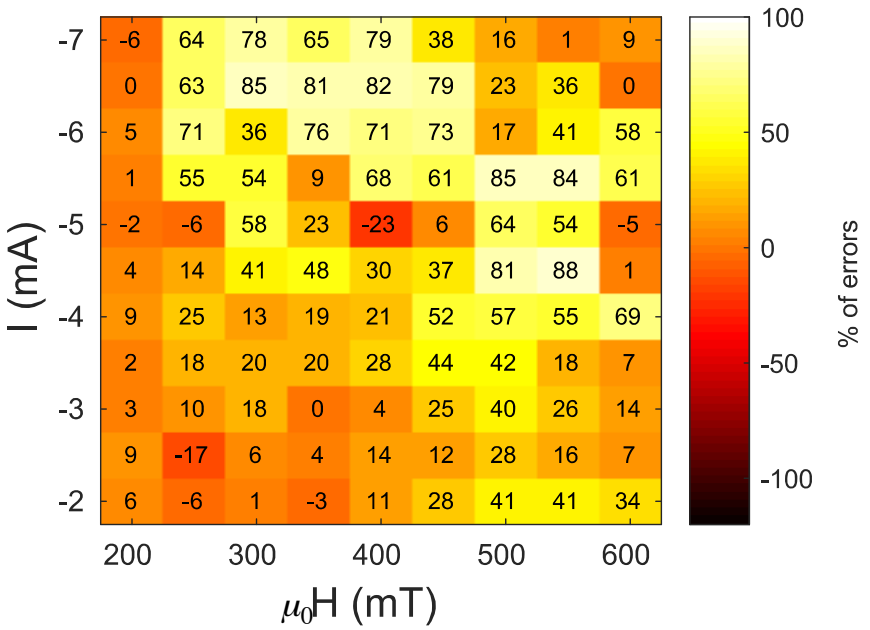

Figure 97: Reduction of error thanks to the feedback during training phase.

the cases during the training phase (Figure 97). But the final error rate is improved in only 66 % of the operating points, during the training phase (Figure 96), which means that feedback has a positive effect on memory, but may also bring unwanted effects which scatter the benefit of the memory in 24 % of cases. The memory can work against the non-linearity in some particular bias points, leading to a

worse classification. At this point, two main questions arise. The first is why doesn't feedback effect decrease evenly the error, for all the operating points. The second question is why does the feedback bring new errors in some cases. In the next section, we study the errors without feedback, to see how the initial situation (without feedback) affects the classification improvement engendered by the feedback.

12.2 INFLUENCE OF THE MEMORY IN THE ERROR RATE WITHOUT FEEDBACK

As seen in the previous chapters, the feedback brings memory and allows separating si3 and si7 cases from sq cases. We chose to quantify the relative reduction of the error when the feedback is added. The initial situation, without feedback, influences the improvement that can be expected. First, the proportion of the errors that are due to misclassified si3 and si7 (Figure 98) should be quantified, because they give the maximum of the error reduction that feedback can provide. This proportion Pr_1 is computed as:

$$Pr_1 = \frac{E_{si3}^{train} + E_{si7}^{train}}{E_{tot}^{train}} \tag{83}$$

where E_{si3}^{train} (respectively E_{si7}^{train}) is the number of misclassified si3 (respectively si7) case during training phase and E_{tot}^{train} is the total number of errors during training phase. As it can be observed, misclassified si3 and si7 input cases always represent the majority of the errors during training phase, but it varies significantly (from $Pr_1 = 51\%$ to $Pr_1 = 100\%$ of the errors) depending on the operating point. Notably for low field and low d.c. current this proportion is between 51% and 78%. For the two particular operating points 600 mT, -4.5 mA and 600 mT, -5 mA, lack of memory is of less importance in the error. It is important to distinguish if this lower proportion denotes that the oscillator classifies correctly a part of si3 and si7 or if other kind of errors arise. In order to answer this question, Figure 99 presents Pr_2 the percentage of si3 and si7 inputs that are correctly classified. This proportion Pr_2 is computed as follows:

$$Pr_2 = \frac{E_{si3}^{train} + E_{si7}^{train}}{N_{si3}^{train} + N_{si7}^{train}} \tag{84}$$

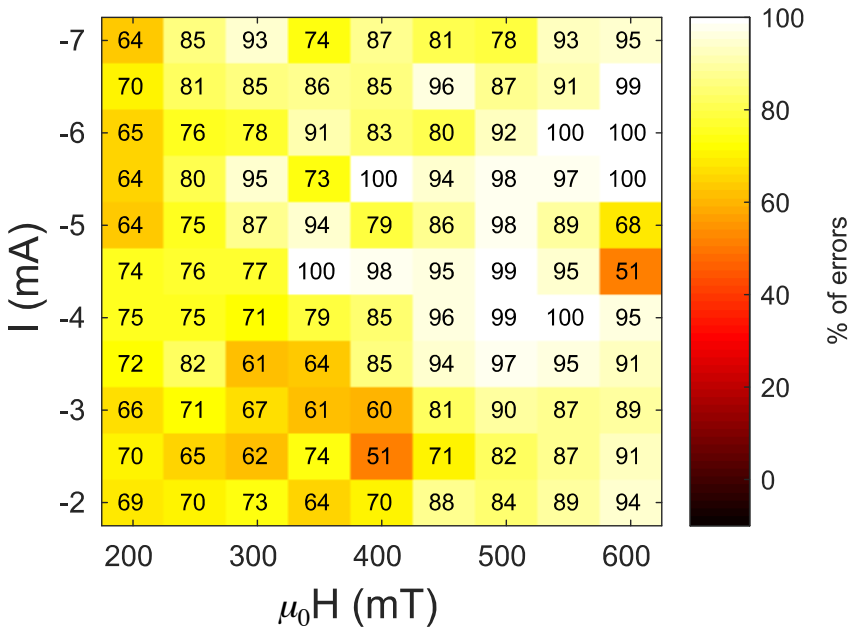

Figure 98: Proportion of misclassified si3 and si7 in the final error Pr_1 versus operating point.

where N_{si3}^{train} (respectively N_{si7}^{train}) is the number of si3 (respectively si7) cases among the training data. For low field and low d.c. current, most of the si3 and si7 cases are misclassified ($Pr_2 > 70$ %), which means that their weaker proportion in the total error ($Pr_1 = 60 - 75$ %) is due to other kind of errors. These other errors come thus from either a lack of separability or a lack of approximation in the reservoir behavior.

For 600 mT, -4.5 mA and 600 mT, -5 mA most of the si3 and si7 cases are well classified ($Pr_2 = 45$ % for 600 mT, -4.5 mA and $Pr_2 = 17$ % for 600 mT, -5 mA). This indicates that the reservoir already possesses a fading memory. Interestingly, other cases present lower error classifying the si3 and si7 (these cases were not suspected because si3 and si7 still represented the most frequent error cases), such as 400 mT, -5.0 mA or 500 mT, -5.5 mA. In these cases, one of the two categories (si3 or si7) is well classified. As a conclusion, without feedback, depending on the operating point, the error distribution is not even. si3 and si7 are still the most common misclassified cases, but for some bias conditions, almost half of the error comes from

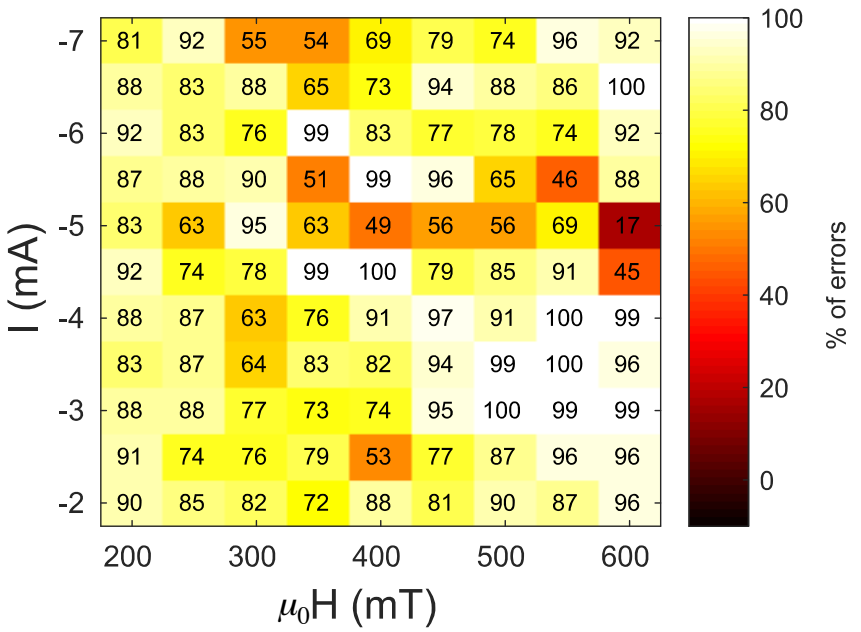

Figure 99: Proportion Pr_2 of si3 and si7 not well classified versus operating point.

other input cases, where lack of memory cannot explain the failure. This lower proportion can either be explained by a larger reservoir memory in some bias condition (such as 600 mT, -4.5 mA and 600 mT, -5 mA) or, by new errors arising, due to a lack of separability or approximation (this is the case at low field and low current). For these conditions, the reduction of the error, by an extrinsic memory, is expected to be smaller. Interestingly, other cases, where the oscillator without feedback exhibits memory, were detected.

12.3 OPERATING POINT DEPENDENT IMPROVEMENT

In this section, we link the physical properties of the oscillator to the observed disparities in the reduction of errors induced by the feedback.

In this section, we analyze the influence of noise in the feedback. Figure 100 plots in a color map the noise level as a function of the bias condition. As in chapter 9, high noise values are observed at low field (200 mT to 400 mT) and low current (-2 mA to -5 mA). This high noise region corresponds also to the main region where

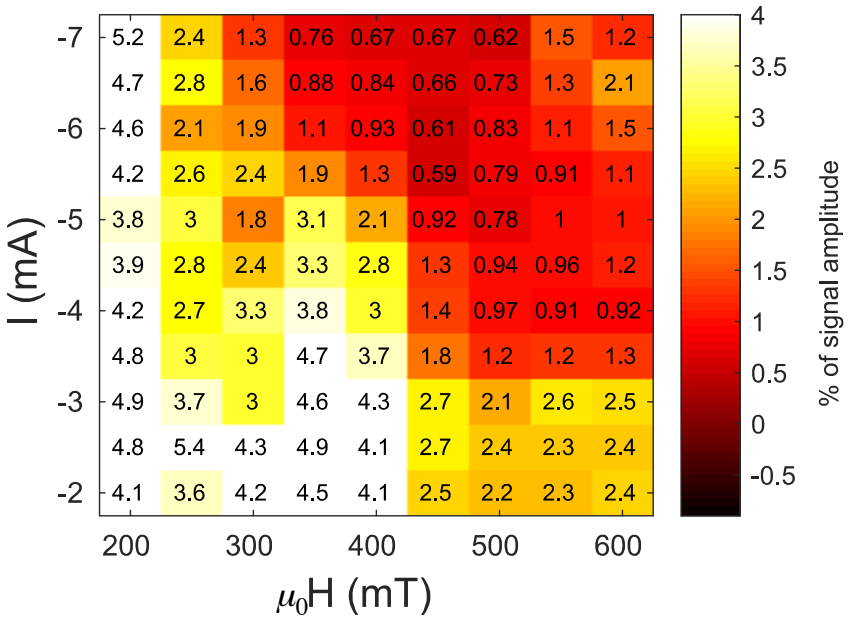

Figure 100: Normalized noise level as a function of the operating point: the magnetic field is swept from 200 mT to 600 mT and the DC current is swept from -2 mA to -7 mA. At low field (200 mT to 400 mT) and low current (-2 mA to -5 mA) the noise level in the oscillation amplitude response is high.

feedback increases the errors and the errors on si3 and si7 cases are less preponderant in the total error. Indeed, in this region, the noise induces errors because the reservoir state is less reproducible for the same input (failure in approximation property). Also, by increasing the relative size of the cluster of points, a linear separation is harder to find (failure in separation property). In the case with feedback, the number of clusters increases, making a linear separation even harder to find. It results in a increase of the error.

12.4.1 *Relaxation time effect*

As said previously, some cases exhibit lower error rates on si3 and si7 most probably due to a "memory". This memory of the oscillator without feedback comes from the relaxation, which itself depends on the operating point. The experimentally measured relaxation time was presented in chapter 10. As mentioned, for 600 mT, -4.5 mA and 600 mT, -5 mA, si3 and si7 are accurately classified without feedback (only 17 % error on these cases). These particularly good results despite a large θ of 180 ns can be explained by the high relaxation time observed at 600 mT and 6.5 mA (see chapter 10). Indeed, as presented in chapter 9, when the input is sent, it explores a large range of the non-linear behavior of the sample (see for instance Figure 55). Thus, at a d.c. current of -4.5 mA and -5 mA, the oscillator can both benefit from the high relaxation time at -6.5 mA and close to the threshold current. Figure 101 shows the time traces in response to si3 input, when compared with the sq cases. As it can be observed, for

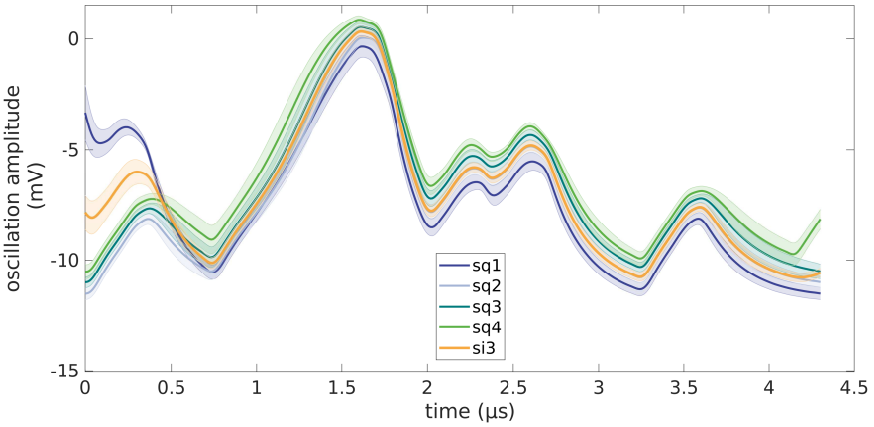

Figure 101: si3 time traces for 600 mT and 5 mA.

the beginning of the curve (time < 0.5 μs), a clear separation between si3 curve and sq cases appears, because at this level of emission $\widetilde{V}$, the relaxation time can reach up to 300 ns. This increase of the relaxation time is only observed for 600 mT (it could be a sign of a magnetization transition from vortex to another mode). This explains the particular results for the 2 bias conditions -4.5 mA and -5.0 mA, observed at 600 mT.

12.4.2 *Conclusion*

The feedback has a different effect depending on the operating point. If in most of the cases it brings memory, it may also generate new errors that overcome the benefits of memory. In general, the feedback is of less interest when the errors without feedback are not completely due to memory. These operating points correspond either to high relaxation time regimes which provide a long intrinsic memory to the oscillator without feedback, or it can correspond to high noise regimes, where the lack of reproducibility degrades the separability of the data. In general, the feedback acts thus better in regimes of high signal over noise ratio and low relaxation time. Also in this study the number of temporal neurons used was relatively moderate (24 temporal neurons). Since the feedback may increase the number of clusters of data (see Chapter 11), it may be easier to find a separation in a higher dimensional state than a 24 dimensions one. Thus a potential perspective would be to improve the number of temporal neurons used in the time-multiplexed reservoir. To stay in a situation where the delay of the line corresponds to the time step τ of the preprocessed input, it would require changing the experimental set-up, with a longer delay line.

CONCLUSION

In this thesis we present the first demonstration of neuromorphic computing with a nanoscale hardware neuron. We use the non-linear transient dynamics of a single nanoscale spin-torque oscillator to emulate the response of a whole neural network, using the reservoir computing approach. The main physical properties used for this demonstration are the non linear dependency of the emitted voltage amplitude with the applied current and the memory of the oscillator through the relaxation of the oscillation amplitude. Using this non-linear dynamics, we classified sine and square waveforms which requires both memory and non-linearity and we recognized successfully digits said by different speakers. For this last task we obtained a recognition rate of 99.6 % using cochlear decomposition prior the oscillator. With a spectrogram as time-to-frequency transformation, the final recognition rate is smaller (80 %) but the oscillator provides a higher gain. The final recognition rate depends on the d.c. current and the magnetic field that is applied to the oscillator, because these bias conditions modify the regime where the oscillator operates and notably the signal to noise ratio as well as the non-linearity of the oscillator response. The noise and the non-linearity are correlated for spin-torque oscillators and optimal results are obtained for intermediate level of noise and non-linearity. The rate and the amplitude of the input which is sent to the oscillator also play an important role. The amplitude of the input influences the amplitude of the signal emitted by the oscillator and generally better results are obtained for larger input signals. The rate should ensure that the oscillator stays in a transient regime while keeping large amplitude variations of the emitted signal. Optimal variation time for the input is found for half of the oscillator relaxation time. Memory can be also added artificially by shifting in time the input signal and the target signal. In the case of spoken digit recognition, the audio waveform needs to be transformed from time-to-frequency domain before being sent to the oscillator. Several transformations were used such as mel frequency cepstral coefficient (MFCC), cochlear decomposition or spectrogram, leading to

different final success rates. Indeed these transformations have various levels of non-linearity, and a high non-linearity transformation such as cochlear decomposition already extracts the important features of the waveforms and can lead to high recognition rates by itself. Adding non-linearity to a spectrogram allows reaching similar success rates. When using these transformations prior to a spin-torque oscillator, higher non-linearity functions give a better final success rate but the benefit of the oscillator is smaller. Thus the role of the time-to-frequency transformations needs to be taken into account in order to interpret the final recognition rate obtained. Finally, we have shown that the memory of the reservoir can be improved by adding a delayed-feedback to the oscillator. The feedback provides a fading memory. The efficiency of this memory for computation was demonstrated on sine/square classification task. Indeed for optimal operating points, the feedback allows suppressing up to 99 % of the errors which were due to a lack of memory. The positive effect of the feedback was observed by visualizing the reservoir states in 2D. The benefit of the feedback closely depends on the operating point and in particular on the relaxation time and on the noise level. It is generally more efficient in regimes of low relaxation times and low noise.

For the purpose of demonstration, we used a single oscillator with time-multiplexing approach to perform recognition. This approach simplified drastically the experimental set-up and allowed to achieve recognition with state of the art results. However for more advanced applications this strategy may suffer of some limitations. First the input needs to be preprocessed in order to drive the non-linear transient dynamics of the spin-torque oscillator. This preprocessing may be a draw-back for real time processing. Moreover the time-multiplexed approach emulates a network with fixed connexions and with neurons connected in a ring shape. For tasks such as image recognition, this architecture may not be the most suited. Finally with time multiplexing, a single oscillator emulates temporal neurons one after the other. Emulating the response of very large neural networks with tens of millions of neurons would required in our case more than a second to provide the output value. Such delay may not be acceptable for critical applications. A promising perspective to overcome these three limitations and take full advantage of the nanoscale size of the spin-torque oscillator is the development of large arrays of intercon-

nected spin-torque oscillators, where each oscillator acts as a single artificial neuron. This strategy also requires to address other issues. Firstly the interconnections of the oscillators should be implemented. Interestingly, spin-torque oscillators can be connected through several mechanisms such as electrical coupling, spin waves or dipolar coupling (see chapter 4), opening different perspectives for the interconnection. For the purpose of recognition, these connections need to have different value and ideally to be reconfigurable.

Diminishing the energy consumption associated to recognition tasks is the motivation for neuromorphic chips. Beside the energy consumption associated to the readout circuitry, the individual energy consumption of each oscillator should also be taken into account. A promising approach to diminish this energy consumption is the use of 10-20 nm large spin-torque oscillators [274]. Indeed, with such structures low currents generate sufficiently high current densities for magnetization gyration.

Finally a fully autonomous chip would require the hardware to perform learning, tuning automatically the coupling between hardware artificial neurons. First experimental proof of learning, changing the coupling between a frequency input and the oscillators was performed in [226] with a small network of four coupled oscillators. Learning could also be implemented using tunable physical connexions that behave like biological synapses. Spintronic memristors are a promising candidate for emulating such functionality [75].

Finally algorithms should be adapted for learning in networks of hardware devices. Current algorithms used for learning in software artificial neural network is the backpropagation which is difficult to transfer to hardware devices, but learning methods using the physics of devices were recently proposed theoretically [275] and may be adapted to an experimental array of interconnected spin-torque oscillators.

This thesis was a first step demonstrating that a single spin-torque oscillator can emulate a neuron at the nanoscale. Associated to these perspectives, this work opens the path to building smart, fast and energy efficient chips able to analyze ambiguous situations in embedded systems.

Part VI

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain.," *Psychological review*, vol. 65, no. 6, p. 386, 1958. (Cited on pages 9, 19, and 34.)

[2] J. Torrejon, M. Riou, F. A. Araujo, S. Tsunegi, G. Khalsa, D. Querlioz, P. Bortolotti, V. Cros, K. Yakushiji, A. Fukushima, *et al.*, "Neuromorphic computing with nanoscale spintronic oscillators," *Nature*, vol. 547, no. 7664, p. 428, 2017. (Cited on pages 10, 20, 113, 145, 149, and 156.)

[3] P. J. Phillips and A. J. O'toole, "Comparison of human and computer performance across face recognition experiments," *Image and Vision Computing*, vol. 32, no. 1, pp. 74–85, 2014. (Cited on page 26.)

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 248–255, Ieee, 2009. (Cited on page 26.)

[5] J. Deng, A. Berg, S. Satheesh, H. Su, A. Khosla, and L. Fei-Fei, "Ilsvrc-2012, 2012," *URL http://www. image-net. org/challenges/LSVRC*, 2012. (Cited on page 26.)

[6] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015. (Cited on page 26.)

[7] C. Thurgood, T. A. Whitfield, and J. Patterson, "Towards a visual recognition threshold: New instrument shows humans identify animals with only 1 ms of visual exposure," *Vision research*, vol. 51, no. 17, pp. 1966–1971, 2011. (Cited on page 26.)

[8] E. R. Kandel, J. H. Schwartz, T. M. Jessell, D. of Biochemistry, M. B. T. Jessell, S. Siegelbaum, and A. Hudspeth, *Principles of neural science*, vol. 4. McGraw-hill New York, 2000. (Cited on page 26.)

[9] L. Lapicque, "Recherches quantitatives sur l'excitation electrique des nerfs traitee comme une polarization," *Journal de Physiologie et de Pathologie Generale*, vol. 9, pp. 620–635, 1907. (Cited on pages 27, 30, and 44.)

[10] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *The Journal of Physiology*, vol. 117, no. 4, pp. 500–544, 1952. (Cited on pages 27, 29, and 30.)

[11] W. M. Kistler, W. Gerstner, and J. L. v. Hemmen, "Reduction of the Hodgkin-Huxley equations to a single-variable threshold model," *Neural computation*, vol. 9, no. 5, pp. 1015–1045, 1997. (Cited on page 30.)

[12] F. Ponulak and A. Kasinski, "Introduction to spiking neural networks: Information processing, learning and applications.," *Acta neurobiologiae experimentalis*, vol. 71, no. 4, pp. 409–433, 2011. (Cited on page 31.)

[13] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015. (Cited on page 33.)

[14] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," tech. rep., California Univ San Diego La Jolla Inst for Cognitive Science, 1985. (Cited on page 35.)

[15] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. (Cited on pages 35 and 36.)

[16] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning.," in *AAAI*, vol. 4, p. 12, 2017. (Cited on page 36.)

[17] A. Canziani, A. Paszke, and E. Culurciello, "An analysis of deep neural network models for practical applications," *arXiv preprint arXiv:1605.07678*, 2016. (Cited on pages 36, 38, and 39.)

[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997. (Cited on page 37.)

[19] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997. (Cited on page 37.)

[20] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, speech and signal processing (icassp), 2013 IEEE International Conference on*, pp. 6645–6649, IEEE, 2013. (Cited on page 37.)

[21] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *arXiv preprint arXiv:1506.00019*, 2015. (Cited on page 37.)

[22] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "Achieving human parity in conversational speech recognition," *arXiv preprint arXiv:1610.05256*, 2016. (Cited on page 37.)

[23] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994. (Cited on page 37.)

[24] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *International Conference on Machine Learning*, pp. 1310–1318, 2013. (Cited on page 37.)

[25] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990. (Cited on page 37.)

[26] H. Jaeger, "The "echo state" approach to analysing and training recurrent neural networks-with an erratum note," *Bonn, Germany: German National Research Center for Information*

Technology GMD Technical Report, vol. 148, no. 34, p. 13, 2001. (Cited on pages 37 and 89.)

[27] W. Maass, T. Natschläger, and H. Markram, "Real-time computing without stable states: A new framework for neural computation based on perturbations," *Neural computation*, vol. 14, no. 11, pp. 2531–2560, 2002. (Cited on pages 37 and 89.)

[28] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future generation computer systems*, vol. 29, no. 7, pp. 1645–1660, 2013. (Cited on page 38.)

[29] J. Von Neumann, "First Draft of a Report on the EDVAC," *IEEE Annals of the History of Computing*, vol. 15, no. 4, pp. 27–75, 1993. (Cited on page 38.)

[30] J. Backus, *Can programming be liberated from the von Neumann style?: a functional style and its algebra of programs*. ACM, 2007. (Cited on page 38.)

[31] M. Horowitz, "1.1 computing's energy problem (and what we can do about it)," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*, pp. 10–14, IEEE, 2014. (Cited on page 39.)

[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012. (Cited on page 39.)

[33] E. Nurvitadhi, G. Venkatesh, J. Sim, D. Marr, R. Huang, J. Ong Gee Hock, Y. T. Liew, K. Srivatsan, D. Moss, S. Subhaschandra, *et al.*, "Can fpgas beat gpus in accelerating next-generation deep neural networks?," in *Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, pp. 5–14, ACM, 2017. (Cited on page 39.)

[34] C. Mead and M. Ismail, *Analog VLSI implementation of neural systems*, vol. 80. Springer Science & Business Media, 2012. (Cited on pages 41 and 43.)

[35] C. Mead, "Neuromorphic electronic systems," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1629–1636, 1990. (Cited on page 41.)

[36] E. Stromatias, F. Galluppi, C. Patterson, and S. Furber, "Power analysis of large-scale, real-time neural networks on spinnaker," in *Neural Networks (IJCNN), The 2013 International Joint Conference on*, pp. 1–8, IEEE, 2013. (Cited on pages 41, 42, and 44.)

[37] B. V. Benjamin, P. Gao, E. McQuinn, S. Choudhary, A. R. Chandrasekaran, J.-M. Bussat, R. Alvarez-Icaza, J. V. Arthur, P. A. Merolla, and K. Boahen, "Neurogrid: A mixed-analog-digital multichip system for large-scale neural simulations," *Proceedings of the IEEE*, vol. 102, no. 5, pp. 699–716, 2014. (Cited on pages 41, 42, and 44.)

[38] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura, *et al.*, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, 2014. (Cited on pages 41, 42, 45, and 46.)

[39] M. Davies, N. Srinivasa, T.-H. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi, N. Imam, S. Jain, *et al.*, "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, 2018. (Cited on pages 41, 42, 45, and 46.)

[40] M. Mahowald and R. Douglas, "A silicon neuron," *Nature*, vol. 354, no. 6354, p. 515, 1991. (Cited on pages 42 and 43.)

[41] G. Indiveri, B. Linares-Barranco, T. J. Hamilton, A. Van Schaik, R. Etienne-Cummings, T. Delbruck, S.-C. Liu, P. Dudek, P. Häfliger, S. Renaud, *et al.*, "Neuromorphic silicon neuron circuits," *Frontiers in neuroscience*, vol. 5, p. 73, 2011. (Cited on pages 42 and 43.)

[42] N. Retdian, S. Takagi, and N. Fujii, "Voltage controlled ring oscillator with wide tuning range and fast voltage swing," in

ASIC, 2002. Proceedings. 2002 IEEE Asia-Pacific Conference on, pp. 201–204, IEEE, 2002. (Cited on pages 42 and 43.)

[43] W. S. T. Yan and H. C. Luong, "A 900-mhz cmos low-phase-noise voltage-controlled ring oscillator," *IEEE Transactions on circuits and systems II: analog and digital signal processing*, vol. 48, no. 2, pp. 216–221, 2001. (Cited on pages 42 and 43.)

[44] X. Zhao, R. Chebli, and M. Sawan, "A wide tuning range voltage-controlled ring oscillator dedicated to ultrasound transmitter," in *Microelectronics, 2004. ICM 2004 Proceedings. The 16th International Conference on*, pp. 313–316, IEEE, 2004. (Cited on page 42.)

[45] A. A. A. Ghani and A. Saparon, "A 1.4 ghz cmos low-phase noise voltage-controlled ring oscillator," in *Research and Development, 2007. SCOReD 2007. 5th Student Conference on*, pp. 1–5, IEEE, 2007. (Cited on page 42.)

[46] Y. A. Eken and J. P. Uyemura, "A 5.9-ghz voltage-controlled ring oscillator in 0.18-/spl mu/m cmos," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 1, pp. 230–233, 2004. (Cited on page 42.)

[47] M. Grozing, B. Phillip, and M. Berroth, "Cmos ring oscillator with quadrature outputs and 100 mhz to 3.5 ghz tuning range," in *Solid-State Circuits Conference, 2003. ESSCIRC'03. Proceedings of the 29th European*, pp. 679–682, IEEE, 2003. (Cited on page 42.)

[48] D. X. Tran and T. T. Dang, "An ultra-low power consumption and very compact 1.49 ghz cmos voltage controlled ring oscillator," in *Advanced Technologies for Communications (ATC), 2014 International Conference on*, pp. 239–244, IEEE, 2014. (Cited on pages 42, 43, and 46.)

[49] Y. Tomita, K. Suzuki, T. Matsumoto, T. Yamamoto, H. Yamaguchi, and H. Tamura, "An 8-to-16ghz 28nm cmos clock distribution circuit based on mutual-injection-locked ring oscillators," in *VLSI Circuits (VLSIC), 2013 Symposium on*, pp. C238–C239, IEEE, 2013. (Cited on pages 42 and 43.)

[50] E. Chicca, F. Stefanini, C. Bartolozzi, and G. Indiveri, "Neuromorphic electronic circuits for building autonomous cognitive systems," *Proceedings of the IEEE*, vol. 102, no. 9, pp. 1367–1388, 2014. (Cited on page 43.)

[51] D. Dupeyron, S. Le Masson, Y. Deval, G. Le Masson, and J.-P. Dom, "A bicmos implementation of the hodgkin-huxley formalism," in *microneuro*, p. 311, IEEE, 1996. (Cited on page 43.)

[52] L. Alvado, J. Tomas, S. Saıghi, S. Renaud, T. Bal, A. Destexhe, and G. Le Masson, "Hardware computation of conductance-based neuron models," *Neurocomputing*, vol. 58, pp. 109–115, 2004. (Cited on page 43.)

[53] M. F. Simoni, G. S. Cymbalyuk, M. E. Sorensen, R. L. Calabrese, and S. P. DeWeerth, "A multiconductance silicon neuron with biologically matched dynamics," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 2, pp. 342–354, 2004. (Cited on page 43.)

[54] T. Yu and G. Cauwenberghs, "Analog vlsi biophysical neurons and synapses with programmable membrane channel kinetics," *IEEE Transactions on Biomedical circuits and Systems*, vol. 4, no. 3, pp. 139–148, 2010. (Cited on page 43.)

[55] G. Indiveri, "A low-power adaptive integrate-and-fire neuron circuit," in *Circuits and Systems, 2003. ISCAS'03. Proceedings of the 2003 International Symposium on*, vol. 4, pp. IV–IV, IEEE, 2003. (Cited on page 43.)

[56] S. Schultz and M. Jabri, "Analogue vlsi'integrate-and-fire'neuron with frequency adaptation," *Electronics Letters*, vol. 31, no. 16, pp. 1357–1358, 1995. (Cited on page 43.)

[57] K. A. Boahen, *Retinomorphic vision systems: Reverse engineering the vertebrate retina*. PhD thesis, California Institute of Technology, 1997. (Cited on page 43.)

[58] A. van Schaik, "Building blocks for electronic spiking neural networks," *Neural networks*, vol. 14, no. 6-7, pp. 617–628, 2001. (Cited on page 43.)

[59] N. Qiao and G. Indiveri, "Scaling mixed-signal neuromorphic processors to 28 nm fd-soi technologies," in *Biomedical Circuits and Systems Conference (BioCAS), 2016 IEEE*, pp. 552–555, IEEE, 2016. (Cited on pages 43 and 44.)

[60] E. M. Izhikevich, "Simple model of spiking neurons," *IEEE Transactions on neural networks*, vol. 14, no. 6, pp. 1569–1572, 2003. (Cited on pages 43 and 44.)

[61] R. Brette and W. Gerstner, "Adaptive exponential integrate-and-fire model as an effective description of neuronal activity," *Journal of neurophysiology*, vol. 94, no. 5, pp. 3637–3642, 2005. (Cited on page 43.)

[62] C. Morris and H. Lecar, "Voltage oscillations in the barnacle giant muscle fiber," *Biophysical journal*, vol. 35, no. 1, pp. 193–213, 1981. (Cited on page 43.)

[63] J. H. Wijekoon and P. Dudek, "Compact silicon neuron circuit with spiking and bursting behaviour," *Neural Networks*, vol. 21, no. 2-3, pp. 524–534, 2008. (Cited on pages 43 and 44.)

[64] A. van Schaik, C. Jin, A. McEwan, and T. J. Hamilton, "A log-domain implementation of the izhikevich neuron model," in *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, pp. 4253–4256, IEEE, 2010. (Cited on page 43.)

[65] V. Rangan, A. Ghosh, V. Aparin, and G. Cauwenberghs, "A subthreshold avlsi implementation of the izhikevich simple neuron model," in *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pp. 4164–4167, IEEE, 2010. (Cited on page 43.)

[66] J. Schemmel, J. Fieres, and K. Meier, "Wafer-scale integration of analog neural networks," in *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, pp. 431–438, IEEE, 2008. (Cited on page 43.)

[67] I. Sourikopoulos, S. Hedayat, C. Loyez, F. Danneville, V. Hoel, E. Mercier, and A. Cappy, "A 4-fj/spike artificial neuron in 65

nm cmos technology," *Frontiers in neuroscience*, vol. 11, p. 123, 2017. (Cited on page 43.)

[68] J. Schemmel, D. Briiderle, A. Griibl, M. Hock, K. Meier, and S. Millner, "A wafer-scale neuromorphic hardware system for large-scale neural modeling," in *Circuits and systems (ISCAS), proceedings of 2010 IEEE international symposium on*, pp. 1947–1950, IEEE, 2010. (Cited on page 44.)

[69] J. Schemmel, A. Grübl, S. Hartmann, A. Kononov, C. Mayr, K. Meier, S. Millner, J. Partzsch, S. Schiefer, S. Scholze, *et al.*, "Live demonstration: A scaled-down version of the brainscales wafer-scale neuromorphic system," in *Circuits and systems (ISCAS), 2012 IEEE international symposium on*, pp. 702–702, IEEE, 2012. (Cited on page 44.)

[70] M. Hopkins, G. Pineda-García, P. A. Bogdan, and S. B. Furber, "Spiking neural networks for computer vision," *Interface Focus*, vol. 8, no. 4, p. 20180007, 2018. (Cited on page 44.)

[71] L. O. Chua and S. M. Kang, "Memristive devices and systems," *Proceedings of the IEEE*, vol. 64, no. 2, pp. 209–223, 1976. (Cited on page 46.)

[72] A. Chanthbouala, V. Garcia, R. O. Cherifi, K. Bouzehouane, S. Fusil, X. Moya, S. Xavier, H. Yamada, C. Deranlot, N. D. Mathur, *et al.*, "A ferroelectric memristor," *Nature materials*, vol. 11, no. 10, p. 860, 2012. (Cited on page 47.)

[73] S. Boyn, J. Grollier, G. Lecerf, B. Xu, N. Locatelli, S. Fusil, S. Girod, C. Carrétéro, K. Garcia, S. Xavier, *et al.*, "Learning through ferroelectric domain dynamics in solid-state synapses," *Nature communications*, vol. 8, p. 14736, 2017. (Cited on page 47.)

[74] A. Chanthbouala, R. Matsumoto, J. Grollier, V. Cros, A. Anane, A. Fert, A. Khvalkovskiy, K. Zvezdin, K. Nishimura, Y. Nagamine, *et al.*, "Vertical-current-induced domain-wall motion in mgo-based magnetic tunnel junctions with low current densities," *Nature Physics*, vol. 7, no. 8, p. 626, 2011. (Cited on page 47.)

[75] S. Lequeux, J. Sampaio, V. Cros, K. Yakushiji, A. Fukushima, R. Matsumoto, H. Kubota, S. Yuasa, and J. Grollier, "A magnetic synapse: multilevel spin-torque memristor with perpendicular anisotropy," *Scientific reports*, vol. 6, p. 31510, 2016. (Cited on pages 47 and 207.)

[76] S. H. Jo, T. Chang, I. Ebong, B. B. Bhadviya, P. Mazumder, and W. Lu, "Nanoscale memristor device as synapse in neuromorphic systems," *Nano letters*, vol. 10, no. 4, pp. 1297–1301, 2010. (Cited on page 47.)

[77] M. Prezioso, F. Merrikh-Bayat, B. Hoskins, G. Adam, K. K. Likharev, and D. B. Strukov, "Training and operation of an integrated neuromorphic network based on metal-oxide memristors," *Nature*, vol. 521, no. 7550, p. 61, 2015. (Cited on page 47.)

[78] G. W. Burr, R. M. Shelby, S. Sidler, C. Di Nolfo, J. Jang, I. Boybat, R. S. Shenoy, P. Narayanan, K. Virwani, E. U. Giacometti, *et al.*, "Experimental demonstration and tolerancing of a large-scale neural network (165 000 synapses) using phase-change memory as the synaptic weight element," *IEEE Transactions on Electron Devices*, vol. 62, no. 11, pp. 3498–3507, 2015. (Cited on page 47.)

[79] S. Ambrogio, P. Narayanan, H. Tsai, R. M. Shelby, I. Boybat, C. Nolfo, S. Sidler, M. Giordano, M. Bodini, N. C. Farinha, *et al.*, "Equivalent-accuracy accelerated neural-network training using analogue memory," *Nature*, vol. 558, no. 7708, p. 60, 2018. (Cited on page 47.)

[80] M. D. Pickett, G. Medeiros-Ribeiro, and R. S. Williams, "A scalable neuristor built with mott memristors," *Nature materials*, vol. 12, no. 2, p. 114, 2013. (Cited on pages 47, 48, and 49.)

[81] A. Beaumont, J. Leroy, J.-C. Orlianges, and A. Crunteanu, "Current-induced electrical self-oscillations across out-of-plane threshold switches based on vo2 layers integrated in crossbars geometry," *Journal of Applied Physics*, vol. 115, no. 15, p. 154502, 2014. (Cited on pages 47, 49, 50, and 51.)

[82] H.-T. Kim, B.-J. Kim, S. Choi, B.-G. Chae, Y. W. Lee, T. Driscoll, M. M. Qazilbash, and D. Basov, "Electrical oscillations induced by the metal-insulator transition in vo 2," *Journal of Applied Physics*, vol. 107, no. 2, p. 023702, 2010. (Cited on pages 47 and 49.)

[83] N. Shukla, A. Parihar, E. Freeman, H. Paik, G. Stone, V. Narayanan, H. Wen, Z. Cai, V. Gopalan, R. Engel-Herbert, *et al.*, "Synchronized charge oscillations in correlated electron systems," *Scientific reports*, vol. 4, p. 4964, 2014. (Cited on pages 47 and 49.)

[84] W. Yi, K. K. Tsang, S. K. Lam, X. Bai, J. A. Crowell, and E. A. Flores, "Biological plausibility and stochasticity in scalable vo 2 active memristor neurons," *Nature communications*, vol. 9, no. 1, p. 4661, 2018. (Cited on pages 47, 49, and 51.)

[85] A. A. Sharma, J. A. Bain, and J. A. Weldon, "Phase coupling and control of oxide-based oscillators for neuromorphic computing," *IEEE Journal on Exploratory Solid-State Computational Devices and Circuits*, vol. 1, pp. 58–66, 2015. (Cited on pages 47, 49, 50, and 51.)

[86] C. Du, F. Cai, M. A. Zidan, W. Ma, S. H. Lee, and W. D. Lu, "Reservoir computing using dynamic memristors for temporal information processing," *Nature communications*, vol. 8, no. 1, p. 2204, 2017. (Cited on pages 47, 50, and 51.)

[87] Z. Wang, S. Joshi, S. Savel'ev, W. Song, R. Midya, Y. Li, M. Rao, P. Yan, S. Asapu, Y. Zhuo, *et al.*, "Fully memristive neural networks for pattern classification with unsupervised learning," *Nature Electronics*, vol. 1, no. 2, p. 137, 2018. (Cited on pages 47, 48, 50, and 51.)

[88] M. D. Pickett and R. S. Williams, "Sub-100 fj and sub-nanosecond thermally driven threshold switching in niobium oxide crosspoint nanodevices," *Nanotechnology*, vol. 23, no. 21, p. 215202, 2012. (Cited on pages 47, 48, 49, and 51.)

[89] N. F. Mott and L. Friedman, "Metal-insulator transitions in vo2, ti2o3 and ti2-x v x o3," *Philosophical Magazine*, vol. 30, no. 2, pp. 389–402, 1974. (Cited on page 47.)

[90] P. Stoliar, L. Cario, E. Janod, B. Corraze, C. Guillot-Deudon, S. Salmon-Bourmand, V. Guiot, J. Tranchant, and M. Rozenberg, "Universal electric-field-driven resistive transition in narrow-gap mott insulators," *Advanced materials*, vol. 25, no. 23, pp. 3222–3226, 2013. (Cited on page 48.)

[91] A. Mehonic and A. J. Kenyon, "Emulating the electrical activity of the neuron using a silicon oxide rram cell," *Frontiers in neuroscience*, vol. 10, p. 57, 2016. (Cited on page 48.)

[92] H. D. Crane, "Neuristor-a novel device and system concept," *Proceedings of the IRE*, vol. 50, no. 10, pp. 2048–2060, 1962. (Cited on page 49.)

[93] L. Wan, M. Zeiler, S. Zhang, Y. Le Cun, and R. Fergus, "Regularization of neural networks using dropconnect," in *International Conference on Machine Learning*, pp. 1058–1066, 2013. (Cited on page 50.)

[94] B. D. Josephson, "Possible new effects in superconductive tunnelling," *Physics letters*, vol. 1, no. 7, pp. 251–253, 1962. (Cited on page 51.)

[95] T. Nagatsuma, K. Enpuku, F. Irie, and K. Yoshida, "Flux-flow type josephson oscillator for millimeter and submillimeter wave region," *Journal of Applied Physics*, vol. 54, no. 6, pp. 3302–3309, 1983. (Cited on page 51.)

[96] A. K. Jain, K. Likharev, J. Lukens, and J. Sauvageau, "Mutual phase-locking in josephson junction arrays," *Physics Reports*, vol. 109, no. 6, pp. 309–426, 1984. (Cited on page 51.)

[97] S. P. Benz and C. J. Burroughs, "Coherent emission from two-dimensional josephson junction arrays," *Applied physics letters*, vol. 58, no. 19, pp. 2162–2164, 1991. (Cited on page 51.)

[98] S. P. Benz and C. J. Burroughs, "Two-dimensional arrays of josephson junctions as voltage-tunable oscillators," *Superconduc-*

tor Science and Technology, vol. 4, no. 11, p. 561, 1991. (Cited on page 51.)

[99] S. P. Benz and P. Booi, "High-frequency oscillators using phase-locked arrays of josephson junctions," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 42, no. 5, pp. 964–966, 1995. (Cited on page 51.)

[100] K. Wiesenfeld, P. Colet, and S. H. Strogatz, "Frequency locking in josephson arrays: Connection with the kuramoto model," *Physical Review E*, vol. 57, no. 2, p. 1563, 1998. (Cited on page 51.)

[101] Y. N. Ovchinnikov and V. Z. Kresin, "Networks of josephson junctions and their synchronization," *Physical Review B*, vol. 88, no. 21, p. 214504, 2013. (Cited on page 51.)

[102] T. Kashiwagi, H. Kubo, K. Sakamoto, T. Yuasa, Y. Tanabe, C. Watanabe, T. Tanaka, Y. Komori, R. Ota, G. Kuwano, *et al.*, "The present status of high-t c superconducting terahertz emitters," *Superconductor Science and Technology*, vol. 30, no. 7, p. 074008, 2017. (Cited on page 51.)

[103] P. Crotty, D. Schult, and K. Segall, "Josephson junction simulation of neurons," *Physical Review E*, vol. 82, no. 1, p. 011914, 2010. (Cited on page 51.)

[104] K. Segall, M. LeGro, S. Kaplan, O. Svitelskiy, S. Khadka, P. Crotty, and D. Schult, "Synchronization dynamics on the picosecond time scale in coupled josephson junction neurons," *Physical Review E*, vol. 95, no. 3, p. 032220, 2017. (Cited on pages 51 and 52.)

[105] D. K. Brock, E. K. Track, and J. M. Rowell, "Superconductor ics: the 100-ghz second generation," *IEEE spectrum*, vol. 37, no. 12, pp. 40–46, 2000. (Cited on page 52.)

[106] H. G. Craighead, "Nanoelectromechanical systems," *Science*, vol. 290, no. 5496, pp. 1532–1535, 2000. (Cited on page 52.)

[107] X. Feng, C. White, A. Hajimiri, and M. L. Roukes, "A self-sustaining ultrahigh-frequency nanoelectromechanical oscillator,"

Nature nanotechnology, vol. 3, no. 6, p. 342, 2008. (Cited on page 53.)

[108] X. M. H. Huang, C. A. Zorman, M. Mehregany, and M. L. Roukes, "Nanoelectromechanical systems: Nanodevice motion at microwave frequencies," *Nature*, vol. 421, no. 6922, p. 496, 2003. (Cited on page 53.)

[109] V. Sazonova, Y. Yaish, H. Üstünel, D. Roundy, T. A. Arias, and P. L. McEuen, "A tunable carbon nanotube electromechanical oscillator," *Nature*, vol. 431, no. 7006, p. 284, 2004. (Cited on page 53.)

[110] M. Zhang, G. S. Wiederhecker, S. Manipatruni, A. Barnard, P. McEuen, and M. Lipson, "Synchronization of micromechanical oscillators using light," *Physical review letters*, vol. 109, no. 23, p. 233906, 2012. (Cited on page 53.)

[111] M. H. Matheny, M. Grau, L. G. Villanueva, R. B. Karabalin, M. Cross, and M. L. Roukes, "Phase synchronization of two anharmonic nanomechanical oscillators," *Physical review letters*, vol. 112, no. 1, p. 014101, 2014. (Cited on page 53.)

[112] B. Ilic, H. Craighead, S. Krylov, W. Senaratne, C. Ober, and P. Neuzil, "Attogram detection using nanoelectromechanical oscillators," *Journal of Applied Physics*, vol. 95, no. 7, pp. 3694–3703, 2004. (Cited on page 53.)

[113] M. Li, H. X. Tang, and M. L. Roukes, "Ultra-sensitive nems-based cantilevers for sensing, scanned probe and very high-frequency applications," *Nature nanotechnology*, vol. 2, no. 2, p. 114, 2007. (Cited on page 53.)

[114] C.-C. Nguyen, "Frequency-selective mems for miniaturized low-power communication devices," *IEEE Transactions on Microwave Theory and Techniques*, vol. 47, no. 8, pp. 1486–1503, 1999. (Cited on page 53.)

[115] J. C. Coulombe, M. C. York, and J. Sylvestre, "Computing with networks of nonlinear mechanical oscillators," *PloS one*, vol. 12, no. 6, p. e0178663, 2017. (Cited on page 53.)

[116] G. Dion, S. Mejaouri, and J. Sylvestre, "Reservoir computing with a single delay-coupled non-linear mechanical oscillator," *Journal of Applied Physics*, vol. 124, no. 15, p. 152132, 2018. (Cited on page 53.)

[117] N. F. Mott, "The electrical conductivity of transition metals," *Proc. R. Soc. Lond. A*, vol. 153, no. 880, pp. 699–717, 1936. (Cited on page 57.)

[118] D. C. Ralph and M. D. Stiles, "Spin transfer torques," *Journal of Magnetism and Magnetic Materials*, vol. 320, no. 7, pp. 1190–1216, 2008. (Cited on pages 57, 64, and 65.)

[119] J. Coey, "Permanent magnets: Plugging the gap," *Scripta Materialia*, vol. 67, no. 6, pp. 524–529, 2012. (Cited on page 57.)

[120] R. Lebrun, *Coupled vortex dynamics in spin-torque oscillators: from resonant excitation to mutual synchronization*. PhD thesis, Paris Saclay, 2015. (Cited on pages 58, 65, and 81.)

[121] M. N. Baibich, J. M. Broto, A. Fert, F. N. Van Dau, F. Petroff, P. Etienne, G. Creuzet, A. Friederich, and J. Chazelas, "Giant magnetoresistance of (001) fe/(001) cr magnetic superlattices," *Physical review letters*, vol. 61, no. 21, p. 2472, 1988. (Cited on pages 59 and 61.)

[122] G. Binasch, P. Grünberg, F. Saurenbach, and W. Zinn, "Enhanced magnetoresistance in layered magnetic structures with antiferromagnetic interlayer exchange," *Physical review B*, vol. 39, no. 7, p. 4828, 1989. (Cited on pages 59 and 61.)

[123] P. M. Levy, S. Zhang, and A. Fert, "Electrical conductivity of magnetic multilayered structures," *Physical review letters*, vol. 65, no. 13, p. 1643, 1990. (Cited on page 61.)

[124] B. Dieny, C. Cowache, A. Nossov, P. Dauguet, J. Chaussy, and P. Gandit, "Anisotropy and angular variation of the giant magnetoresistance in magnetic multilayers," *Journal of applied physics*, vol. 79, no. 8, pp. 6370–6375, 1996. (Cited on page 61.)

[125] J. Daughton, "Gmr applications," *Journal of Magnetism and Magnetic Materials*, vol. 192, no. 2, pp. 334–342, 1999. (Cited on page 61.)

[126] T. Valet and A. Fert, "Classical theory of perpendicular giant magnetoresistance in magnetic multilayers," *Journal of Magnetism and Magnetic Materials*, vol. 121, no. 1-3, pp. 378–382, 1993. (Cited on page 61.)

[127] A. Berkowitz, J. Mitchell, M. Carey, A. Young, S. Zhang, F. Spada, F. Parker, A. Hutten, and G. Thomas, "Giant magnetoresistance in heterogeneous cu-co alloys," *Physical Review Letters*, vol. 68, no. 25, p. 3745, 1992. (Cited on page 61.)

[128] S. Mangin, D. Ravelosona, J. Katine, M. Carey, B. Terris, and E. E. Fullerton, "Current-induced magnetization reversal in nanopillars with perpendicular anisotropy," *Nature materials*, vol. 5, no. 3, p. 210, 2006. (Cited on page 61.)

[129] D. K. Pandya, P. Gupta, S. C. Kashyap, and S. Chaudhary, "Gmr in excess of 10% at room temperature and low magnetic fields in electrodeposited cu/co nano-multilayer structures," *MRS Online Proceedings Library Archive*, vol. 961, 2006. (Cited on page 61.)

[130] M. Julliere, "Tunneling between ferromagnetic films," *Physics letters A*, vol. 54, no. 3, pp. 225–226, 1975. (Cited on pages 61 and 63.)

[131] T. Miyazaki and N. Tezuka, "Giant magnetic tunneling effect in fe/al2o3/fe junction," *Journal of Magnetism and Magnetic Materials*, vol. 139, no. 3, pp. L231–L234, 1995. (Cited on pages 61 and 63.)

[132] J. S. Moodera, L. R. Kinder, T. M. Wong, and R. Meservey, "Large magnetoresistance at room temperature in ferromagnetic thin film tunnel junctions," *Physical review letters*, vol. 74, no. 16, p. 3273, 1995. (Cited on pages 61 and 63.)

[133] E. Y. Tsymbal, O. N. Mryasov, and P. R. LeClair, "Spin-dependent tunnelling in magnetic tunnel junctions," *Journal of Physics: Condensed Matter*, vol. 15, no. 4, p. R109, 2003. (Cited on page 61.)

[134] J. Bardeen, "Tunnelling from a many-particle point of view," *Physical Review Letters*, vol. 6, no. 2, p. 57, 1961. (Cited on page 62.)

[135] P. Talatchian, *Bio-inspired computing based on the synchronization of magnetic oscillators*. PhD thesis, Paris Saclay, 2019. (Cited on pages 62, 74, 75, and 86.)

[136] J. C. Slonczewski, "Conductance and exchange coupling of two ferromagnets separated by a tunneling barrier," *Physical Review B*, vol. 39, no. 10, p. 6995, 1989. (Cited on page 63.)

[137] W. Gallagher, S. S. Parkin, Y. Lu, X. Bian, A. Marley, K. Roche, R. Altman, S. Rishton, C. Jahnes, T. Shaw, *et al.*, "Microstructured magnetic tunnel junctions," *Journal of Applied Physics*, vol. 81, no. 8, pp. 3741–3746, 1997. (Cited on page 63.)

[138] W. Butler, X.-G. Zhang, T. Schulthess, and J. MacLaren, "Spin-dependent tunneling conductance of fe| mgo| fe sandwiches," *Physical Review B*, vol. 63, no. 5, p. 054416, 2001. (Cited on page 64.)

[139] S. Yuasa, T. Nagahama, A. Fukushima, Y. Suzuki, and K. Ando, "Giant room-temperature magnetoresistance in single-crystal fe/mgo/fe magnetic tunnel junctions," *Nature materials*, vol. 3, no. 12, p. 868, 2004. (Cited on page 64.)

[140] S. S. Parkin, C. Kaiser, A. Panchula, P. M. Rice, B. Hughes, M. Samant, and S.-H. Yang, "Giant tunnelling magnetoresistance at room temperature with mgo (100) tunnel barriers," *Nature materials*, vol. 3, no. 12, p. 862, 2004. (Cited on page 64.)

[141] P. Freitas and L. Berger, "Observation of s-d exchange force between domain walls and electric current in very thin permalloy films," *Journal of Applied Physics*, vol. 57, no. 4, pp. 1266–1269, 1985. (Cited on page 64.)

[142] C.-Y. Hung and L. Berger, "Exchange forces between domain wall and electric current in permalloy films of variable thickness," *Journal of applied physics*, vol. 63, no. 8, pp. 4276–4278, 1988. (Cited on page 64.)

[143] L. Berger, "Emission of spin waves by a magnetic multilayer traversed by a current," *Physical Review B*, vol. 54, no. 13, p. 9353, 1996. (Cited on page 64.)

[144] J. C. Slonczewski, "Current-driven excitation of magnetic multilayers," *Journal of Magnetism and Magnetic Materials*, vol. 159, no. 1-2, pp. L1–L7, 1996. (Cited on page 64.)

[145] M. D. Stiles and J. Miltat, "Spin-transfer torque and dynamics," in *Spin dynamics in confined magnetic structures III*, pp. 225–308, Springer, 2006. (Cited on pages 64 and 66.)

[146] J. C. Sankey, Y.-T. Cui, J. Z. Sun, J. C. Slonczewski, R. A. Buhrman, and D. C. Ralph, "Measurement of the spin-transfer-torque vector in magnetic tunnel junctions," *Nature Physics*, vol. 4, no. 1, p. 67, 2008. (Cited on page 65.)

[147] C. Wang, Y.-T. Cui, J. A. Katine, R. A. Buhrman, and D. C. Ralph, "Time-resolved measurement of spin-transfer-driven ferromagnetic resonance and spin torque in magnetic tunnel junctions," *Nature Physics*, vol. 7, no. 6, p. 496, 2011. (Cited on page 65.)

[148] M. D. Stiles and A. Zangwill, "Anatomy of spin-transfer torque," *Physical Review B*, vol. 66, no. 1, p. 014407, 2002. (Cited on page 66.)

[149] I. Theodonis, N. Kioussis, A. Kalitsov, M. Chshiev, and W. Butler, "Anomalous bias dependence of spin torque in magnetic tunnel junctions," *Physical review letters*, vol. 97, no. 23, p. 237205, 2006. (Cited on page 67.)

[150] M. Chshiev, I. Theodonis, A. Kalitsov, N. Kioussis, and W. Butler, "Voltage dependence of spin transfer torque in magnetic tunnel junctions," *IEEE Transactions on Magnetics*, vol. 44, no. 11, pp. 2543–2546, 2008. (Cited on page 67.)

[151] S.-C. Oh, S.-Y. Park, A. Manchon, M. Chshiev, J.-H. Han, H.-W. Lee, J.-E. Lee, K.-T. Nam, Y. Jo, Y.-C. Kong, *et al.*, "Bias-voltage dependence of perpendicular spin-transfer torque in asymmetric mgo-based magnetic tunnel junctions," *Nature Physics*, vol. 5, no. 12, p. 898, 2009. (Cited on page 67.)

[152] A. Dussaux, A. V. Khvalkovskiy, P. Bortolotti, J. Grollier, V. Cros, and A. Fert, "Field dependence of spin-transfer-induced vortex dynamics in the nonlinear regime," *Physical Review B*, vol. 86, p. 014402, July 2012. (Cited on page 67.)

[153] E. Grimaldi, A. Dussaux, P. Bortolotti, J. Grollier, G. Pillet, A. Fukushima, H. Kubota, K. Yakushiji, S. Yuasa, and V. Cros, "Response to noise of a vortex based spin transfer nano-oscillator," *Physical Review B*, vol. 89, no. 10, p. 104404, 2014. (Cited on pages 67, 76, 83, 156, 158, 163, and 167.)

[154] L. Landau and E. Lifshitz, "On the theory of the dispersion of magnetic permeability in ferromagnetic bodies," *Phys. Z. Sowjetunion*, vol. 8, no. 153, pp. 101–114, 1935. (Cited on page 68.)

[155] T. L. Gilbert, "A phenomenological theory of damping in ferromagnetic materials," *IEEE Transactions on Magnetics*, vol. 40, no. 6, pp. 3443–3449, 2004. (Cited on page 68.)

[156] S. Tsunegi, H. Kubota, S. Tamaru, K. Yakushiji, M. Konoto, A. Fukushima, T. Taniguchi, H. Arai, H. Imamura, and S. Yuasa, "Damping parameter and interfacial perpendicular magnetic anisotropy of feb nanopillar sandwiched between mgo barrier and cap layers in magnetic tunnel junctions," *Applied Physics Express*, vol. 7, no. 3, p. 033004, 2014. (Cited on page 68.)

[157] A. Slavin and V. Tiberkevich, "Nonlinear auto-oscillator theory of microwave generation by spin-polarized current," *IEEE Transactions on Magnetics*, vol. 45, no. 4, pp. 1875–1918, 2009. (Cited on pages 68, 69, 76, 84, 138, 155, 158, 165, and 167.)

[158] S. Kaka, M. R. Pufall, W. H. Rippard, T. J. Silva, S. E. Russek, and J. A. Katine, "Mutual phase-locking of microwave spin torque nano-oscillators," *Nature*, vol. 437, no. 7057, pp. 389–392, 2005. (Cited on page 72.)

[159] F. B. Mancoff, N. D. Rizzo, B. N. Engel, and S. Tehrani, "Phase-locking in double-point-contact spin-transfer devices," *Nature*, vol. 437, no. 7057, pp. 393–395, 2005. (Cited on page 72.)

[160] N. Locatelli, A. Hamadeh, F. A. Araujo, A. D. Belanovsky, P. N. Skirdkov, R. Lebrun, V. V. Naletov, K. A. Zvezdin, M. Munoz, and J. Grollier, "Efficient synchronization of dipolarly coupled vortex-based spin transfer nano-oscillators," *Scientific reports*, vol. 5, 2015. (Cited on page 72.)

[161] R. Lebrun, S. Tsunegi, P. Bortolotti, H. Kubota, A. S. Jenkins, M. Romera, K. Yakushiji, A. Fukushima, J. Grollier, S. Yuasa, and V. Cros, "Mutual synchronization of spin torque nano-oscillators through a long-range and tunable electrical coupling scheme," *Nature Communications*, vol. 8, p. ncomms15825, June 2017. (Cited on page 72.)

[162] J. Katine, F. Albert, R. Buhrman, E. Myers, and D. Ralph, "Current-driven magnetization reversal and spin-wave excitations in co/cu/co pillars," *Physical review letters*, vol. 84, no. 14, p. 3149, 2000. (Cited on page 72.)

[163] J.-E. Wegrowe, A. Fabian, P. Guittienne, X. Hoffer, D. Kelly, J.-P. Ansermet, and E. Olive, "Exchange torque and spin transfer between spin polarized current and ferromagnetic layers," *Applied physics letters*, vol. 80, no. 20, pp. 3775–3777, 2002. (Cited on page 72.)

[164] J. Sun, D. Monsma, D. Abraham, M. Rooks, and R. Koch, "Batch-fabricated spin-injection magnetic switches," *Applied Physics Letters*, vol. 81, no. 12, pp. 2202–2204, 2002. (Cited on page 72.)

[165] M. Tsoi, A. Jansen, J. Bass, W.-C. Chiang, M. Seck, V. Tsoi, and P. Wyder, "Excitation of a magnetic multilayer by an electric current," *Physical Review Letters*, vol. 80, no. 19, p. 4281, 1998. (Cited on page 72.)

[166] E. Myers, D. Ralph, J. Katine, R. Louie, and R. Buhrman, "Current-induced switching of domains in magnetic multilayer devices," *Science*, vol. 285, no. 5429, pp. 867–870, 1999. (Cited on page 72.)

[167] M. R. Pufall, W. H. Rippard, and T. J. Silva, "Materials dependence of the spin-momentum transfer efficiency and critical

current in ferromagnetic metal/cu multilayers," *Applied physics letters*, vol. 83, no. 2, pp. 323–325, 2003. (Cited on page 72.)

[168] N. Locatelli, *Dynamique par transfert de spin et synchronisation d'oscillateurs couplés à base de vortex magnétiques*. PhD thesis, Paris 11, 2012. (Cited on page 72.)

[169] S. I. Kiselev, J. Sankey, I. Krivorotov, N. Emley, R. Schoelkopf, R. Buhrman, and D. Ralph, "Microwave oscillations of a nanomagnet driven by a spin-polarized current," *nature*, vol. 425, no. 6956, p. 380, 2003. (Cited on page 72.)

[170] W. H. Rippard, M. R. Pufall, S. Kaka, T. J. Silva, and S. E. Russek, "Current-driven microwave dynamics in magnetic point contacts as a function of applied field angle," *Physical Review B*, vol. 70, no. 10, p. 100406, 2004. (Cited on pages 72 and 74.)

[171] V. Pribiag, I. Krivorotov, G. Fuchs, P. Braganca, O. Ozatay, J. Sankey, D. Ralph, and R. Buhrman, "Magnetic vortex oscillator driven by dc spin-polarized current," *Nature Physics*, vol. 3, no. 7, p. 498, 2007. (Cited on pages 73 and 74.)

[172] A. Houshang, M. Fazlali, S. Sani, P. Dürrenfeld, E. Iacocca, J. Åkerman, and R. Dumas, "Effect of excitation fatigue on the synchronization of multiple nanocontact spin-torque oscillators," *IEEE Magnetics Letters*, vol. 5, pp. 1–4, 2014. (Cited on pages 73 and 74.)

[173] S. Tsunegi, K. Yakushiji, A. Fukushima, S. Yuasa, and H. Kubota, "Microwave emission power exceeding 10 μW in spin torque vortex oscillator," *Applied Physics Letters*, vol. 109, no. 25, p. 252402, 2016. (Cited on pages 73, 74, 76, and 105.)

[174] S. Parkin and D. Mauri, "Spin engineering: Direct determination of the ruderman-kittel-kasuya-yosida far-field range function in ruthenium," *Physical Review B*, vol. 44, no. 13, p. 7131, 1991. (Cited on page 73.)

[175] O. Klein, G. De Loubens, V. Naletov, F. Boust, T. Guillet, H. Hurdequint, A. Leksikov, A. Slavin, V. Tiberkevich, and N. Vukadinovic, "Ferromagnetic resonance force spectroscopy of

individual submicron-size samples," *Physical Review B*, vol. 78, no. 14, p. 144410, 2008. (Cited on page 73.)

[176] R. K. Dumas, E. Iacocca, S. Bonetti, S. Sani, S. M. Mohseni, A. Eklund, J. Persson, O. Heinonen, and J. Åkerman, "Spin-wave-mode coexistence on the nanoscale: a consequence of the oersted-field-induced asymmetric energy landscape," *Physical review letters*, vol. 110, no. 25, p. 257202, 2013. (Cited on page 73.)

[177] P. Muduli, O. Heinonen, and J. Åkerman, "Decoherence and mode hopping in a magnetic tunnel junction based spin torque oscillator," *Physical review letters*, vol. 108, no. 20, p. 207203, 2012. (Cited on page 73.)

[178] E. Iacocca, O. Heinonen, P. Muduli, and J. Åkerman, "Generation linewidth of mode-hopping spin torque oscillators," *Physical Review B*, vol. 89, no. 5, p. 054402, 2014. (Cited on page 73.)

[179] M. R. Pufall, W. H. Rippard, M. L. Schneider, and S. E. Russek, "Low-field current-hysteretic oscillations in spin-transfer nanocontacts," *Physical Review B*, vol. 75, no. 14, p. 140404, 2007. (Cited on page 74.)

[180] Q. Mistral, M. van Kampen, G. Hrkac, J.-V. Kim, T. Devolder, P. Crozat, C. Chappert, L. Lagae, and T. Schrefl, "Current-driven vortex oscillations in metallic nanocontacts," *Physical review letters*, vol. 100, no. 25, p. 257201, 2008. (Cited on page 74.)

[181] S. Tsunegi, H. Kubota, K. Yakushiji, M. Konoto, S. Tamaru, A. Fukushima, H. Arai, H. Imamura, E. Grimaldi, R. Lebrun, *et al.*, "High emission power and q factor in spin torque vortex oscillator consisting of feb free layer," *Applied Physics Express*, vol. 7, no. 6, p. 063009, 2014. (Cited on pages 74 and 106.)

[182] S. Bonetti, P. Muduli, F. Mancoff, and J. Åkerman, "Spin torque oscillator frequency versus magnetic field angle: The prospect of operation beyond 65 ghz," *Applied Physics Letters*, vol. 94, no. 10, p. 102507, 2009. (Cited on page 74.)

[183] S. I. Kiselev, J. C. Sankey, I. N. Krivorotov, N. C. Emley, R. J. Schoelkopf, R. A. Buhrman, and D. C. Ralph, "Microwave oscillations of a nanomagnet driven by a spin-polarized current," *Nature*, vol. 425, no. 6956, pp. 380–383, 2003. (Cited on page 74.)

[184] T. Seki, Y. Sakuraba, R. Okura, and K. Takanashi, "High power radio frequency oscillation by spin transfer torque in a co2mnsi layer: Experiment and macrospin simulation," *Journal of Applied Physics*, vol. 113, no. 3, p. 033907, 2013. (Cited on page 74.)

[185] Q. Mistral, J.-V. Kim, T. Devolder, P. Crozat, C. Chappert, J. Katine, M. Carey, and K. Ito, "Current-driven microwave oscillations in current perpendicular-to-plane spin-valve nanopillars," *Applied physics letters*, vol. 88, no. 19, p. 192507, 2006. (Cited on page 74.)

[186] T. Seki, Y. Sakuraba, H. Arai, M. Ueda, R. Okura, H. Imamura, and K. Takanashi, "High power all-metal spin torque oscillator using full heusler co2 (fe, mn) si," *Applied Physics Letters*, vol. 105, no. 9, p. 092406, 2014. (Cited on page 74.)

[187] T. Yamamoto, T. Seki, T. Kubota, H. Yako, and K. Takanashi, "Zero-field spin torque oscillation in co2 (fe, mn) si with a point contact geometry," *Applied Physics Letters*, vol. 106, no. 9, p. 092406, 2015. (Cited on page 74.)

[188] W. H. Rippard, M. R. Pufall, S. Kaka, S. E. Russek, and T. J. Silva, "Direct-current induced dynamics in C o 90 F e 10/N i 80 F e 20 point contacts," *Physical Review Letters*, vol. 92, no. 2, p. 027201, 2004. (Cited on page 74.)

[189] S. Bonetti, P. Muduli, F. Mancoff, and J. \AAkerman, "Spin torque oscillator frequency versus magnetic field angle: The prospect of operation beyond 65 GHz," *Applied Physics Letters*, vol. 94, no. 10, p. 102507, 2009. (Cited on page 74.)

[190] A. V. Nazarov, K. Nikolaev, Z. Gao, H. Cho, and D. Song, "Microwave generation in mgo magnetic tunnel junctions due to spin transfer effects," *Journal of Applied Physics*, vol. 103, no. 7, p. 07A503, 2008. (Cited on page 74.)

[191] D. Houssameddine, J. Sierra, D. Gusakova, B. Delaet, U. Ebels, L. Buda-Prejbeanu, M.-C. Cyrille, B. Dieny, B. Ocker, J. Langer, *et al.*, "Spin torque driven excitations in a synthetic antiferromagnet," *Applied Physics Letters*, vol. 96, no. 7, p. 072511, 2010. (Cited on page 74.)

[192] P. K. Muduli, Y. Pogoryelov, S. Bonetti, G. Consolo, F. Mancoff, and J. Åkerman, "Nonlinear frequency and amplitude modulation of a nanocontact-based spin-torque oscillator," *Physical Review B*, vol. 81, no. 14, p. 140408, 2010. (Cited on page 74.)

[193] W. Rippard, M. Pufall, and A. Kos, "Time required to injection-lock spin torque nanoscale oscillators," *Applied Physics Letters*, vol. 103, no. 18, p. 182403, 2013. (Cited on page 74.)

[194] J. Costa, S. Serrano-Guisan, B. Lacoste, A. Jenkins, T. Böhnert, M. Tarequzzaman, J. Borme, F. Deepak, E. Paz, J. Ventura, *et al.*, "High power and low critical current density spin transfer torque nano-oscillators using mgo barriers with intermediate thickness," *Scientific Reports*, vol. 7, no. 1, p. 7237, 2017. (Cited on page 74.)

[195] H. Kubota, K. Yakushiji, A. Fukushima, S. Tamaru, M. Konoto, T. Nozaki, S. Ishibashi, T. Saruya, S. Yuasa, T. Taniguchi, *et al.*, "Spin-torque oscillator based on magnetic tunnel junction with a perpendicularly magnetized free layer and in-plane magnetized polarizer," *Applied Physics Express*, vol. 6, no. 10, p. 103003, 2013. (Cited on page 74.)

[196] H. Maehara, H. Kubota, Y. Suzuki, T. Seki, K. Nishimura, Y. Nagamine, K. Tsunekawa, A. Fukushima, A. M. Deac, K. Ando, *et al.*, "Large emission power over 2 μw with high q factor obtained from nanocontact magnetic-tunnel-junction-based spin torque oscillator," *Applied Physics Express*, vol. 6, no. 11, p. 113005, 2013. (Cited on page 74.)

[197] H. Maehara, H. Kubota, Y. Suzuki, T. Seki, K. Nishimura, Y. Nagamine, K. Tsunekawa, A. Fukushima, H. Arai, T. Taniguchi, *et al.*, "High q factor over 3000 due to out-of-plane precession in nano-contact spin-torque oscillator based

on magnetic tunnel junctions," *Applied Physics Express*, vol. 7, no. 2, p. 023003, 2014. (Cited on page 74.)

[198] A. Dussaux, B. Georges, J. Grollier, V. Cros, A. V. Khvalkovskiy, A. Fukushima, M. Konoto, H. Kubota, K. Yakushiji, and S. Yuasa, "Large microwave generation from current-driven magnetic vortex oscillators in magnetic tunnel junctions," *Nature communications*, vol. 1, p. 8, 2010. (Cited on page 74.)

[199] O. Boulle, V. Cros, J. Grollier, L. Pereira, C. Deranlot, F. Petroff, G. Faini, J. Barnaś, and A. Fert, "Shaped angular dependence of the spin-transfer torque and microwave generation without magnetic field," *Nature Physics*, vol. 3, no. 7, p. 492, 2007. (Cited on page 75.)

[200] D. Houssameddine, U. Ebels, B. Delaët, B. Rodmacq, I. Firastrau, F. Ponthenier, M. Brunet, C. Thirion, J.-P. Michel, L. Prejbeanu-Buda, *et al.*, "Spin-torque oscillator using a perpendicular polarizer and a planar free layer," *Nature materials*, vol. 6, no. 6, p. 447, 2007. (Cited on page 75.)

[201] Z. Zeng, G. Finocchio, B. Zhang, P. K. Amiri, J. A. Katine, I. N. Krivorotov, Y. Huai, J. Langer, B. Azzerboni, K. L. Wang, *et al.*, "Ultralow-current-density and bias-field-free spin-transfer nano-oscillator," *Scientific reports*, vol. 3, p. srep01426, 2013. (Cited on page 75.)

[202] T. Devolder, A. Meftah, K. Ito, J. Katine, P. Crozat, and C. Chappert, "Spin transfer oscillators emitting microwave in zero applied magnetic field," *Journal of applied physics*, vol. 101, no. 6, p. 063916, 2007. (Cited on page 75.)

[203] A. Khvalkovskiy, J. Grollier, A. Dussaux, K. A. Zvezdin, and V. Cros, "Vortex oscillations induced by spin-polarized current in a magnetic nanopillar: Analytical versus micromagnetic calculations," *Physical Review B*, vol. 80, no. 14, p. 140401, 2009. (Cited on pages 76 and 82.)

[204] N. Locatelli, V. Naletov, J. Grollier, G. De Loubens, V. Cros, C. Deranlot, C. Ulysse, G. Faini, O. Klein, and A. Fert, "Dynamics of two coupled vortices in a spin valve nanopillar excited

by spin transfer torque," *Applied Physics Letters*, vol. 98, no. 6, p. 062501, 2011. (Cited on page 76.)

[205] R. Lebrun, N. Locatelli, S. Tsunegi, J. Grollier, V. Cros, F. A. Araujo, H. Kubota, K. Yakushiji, A. Fukushima, and S. Yuasa, "Nonlinear behavior and mode coupling in spin-transfer nano-oscillators," *Physical Review Applied*, vol. 2, no. 6, p. 061001, 2014. (Cited on page 76.)

[206] A. Hubert and R. Schäfer, *Magnetic domains: the analysis of magnetic microstructures*. Springer Science & Business Media, 2008. (Cited on page 79.)

[207] E. Grimaldi, *Étude des propriétés non-linéaires et de l'origine du bruit d'oscillateurs à transfert de spin à base de vortex : vers le développement de nano-dispositifs radiofréquences spintroniques*. phdthesis, UniversitÃ© Paris Sud - Paris XI, June 2015. (Cited on page 80.)

[208] K. L. Metlov and K. Y. Guslienko, "Stability of magnetic vortex in soft magnetic nano-sized circular cylinder," *Journal of magnetism and magnetic materials*, vol. 242, pp. 1015–1017, 2002. (Cited on page 80.)

[209] K. Y. Guslienko, B. Ivanov, V. Novosad, Y. Otani, H. Shima, and K. Fukamichi, "Eigenfrequencies of vortex state excitations in magnetic submicron-size disks," *Journal of Applied Physics*, vol. 91, no. 10, pp. 8037–8039, 2002. (Cited on pages 82 and 83.)

[210] Y. Gaididei, V. P. Kravchuk, and D. D. Sheka, "Magnetic vortex dynamics induced by an electrical current," *International Journal of Quantum Chemistry*, vol. 110, pp. 83–97, Jan. 2010. (Cited on page 82.)

[211] A. A. Tulapurkar, Y. Suzuki, A. Fukushima, H. Kubota, H. Maehara, K. Tsunekawa, D. D. Djayaprawira, N. Watanabe, and S. Yuasa, "Spin-torque diode effect in magnetic tunnel junctions," *Nature*, vol. 438, no. 7066, pp. 339–342, 2005. (Cited on page 84.)

[212] S. Tamaru, H. Kubota, K. Yakushiji, S. Yuasa, and A. Fukushima, "Extremely coherent microwave emission from

spin torque oscillator stabilized by phase locked loop," *Scientific reports*, vol. 5, p. 18134, 2015. (Cited on page 84.)

[213] M. Kreissig, R. Lebrun, F. Protze, K. Merazzo, J. Hem, L. Vila, R. Ferreira, M. Cyrille, F. Ellinger, V. Cros, *et al.*, "Vortex spin-torque oscillator stabilized by phase locked loop using integrated circuits," *AIP Advances*, vol. 7, no. 5, p. 056653, 2017. (Cited on page 84.)

[214] S. Tsunegi, T. Taniguchi, R. Lebrun, K. Yakushiji, V. Cros, J. Grollier, A. Fukushima, S. Yuasa, and H. Kubota, "Scaling up electrically synchronized spin torque oscillator networks," *Scientific reports*, vol. 8, no. 1, p. 13475, 2018. (Cited on page 84.)

[215] H. S. Choi, S. Y. Kang, S. J. Cho, I.-Y. Oh, M. Shin, H. Park, C. Jang, B.-C. Min, S.-I. Kim, S.-Y. Park, *et al.*, "Spin nano–oscillator–based wireless communication," *Scientific reports*, vol. 4, p. 5486, 2014. (Cited on page 84.)

[216] A. Ruiz-Calaforra, A. Purbawati, T. Brächer, J. Hem, C. Murapaka, E. Jiménez, D. Mauri, A. Zeltser, J. Katine, M.-C. Cyrille, *et al.*, "Frequency shift keying by current modulation in a mtj-based stno with high data rate," *Applied Physics Letters*, vol. 111, no. 8, p. 082401, 2017. (Cited on page 84.)

[217] A. Purbawati, F. Garcia-Sanchez, L. Buda-Prejbeanu, and U. Ebels, "Enhanced modulation rates via field modulation in spin torque nano-oscillators," *Applied Physics Letters*, vol. 108, no. 12, p. 122402, 2016. (Cited on page 84.)

[218] K. Mizushima, K. Kudo, T. Nagasawa, and R. Sato, "High-data-transfer-rate read heads composed of spin-torque oscillators," in *Journal of Physics: Conference Series*, vol. 266, p. 012060, IOP Publishing, 2011. (Cited on page 85.)

[219] J.-G. Zhu, X. Zhu, and Y. Tang, "Microwave assisted magnetic recording," *IEEE Transactions on Magnetics*, vol. 44, no. 1, pp. 125–131, 2008. (Cited on page 85.)

[220] A. Tulapurkar, Y. Suzuki, A. Fukushima, H. Kubota, H. Maehara, K. Tsunekawa, D. Djayaprawira, N. Watanabe, and

S. Yuasa, "Spin-torque diode effect in magnetic tunnel junctions," *Nature*, vol. 438, no. 7066, p. 339, 2005. (Cited on page 85.)

[221] S. Menshawy, A. Jenkins, K. J. Merazzo, L. Vila, R. Ferreira, M.-C. Cyrille, U. Ebels, P. Bortolotti, J. Kermorvant, and V. Cros, "Spin transfer driven resonant expulsion of a magnetic vortex core for efficient rf detector," *AIP Advances*, vol. 7, no. 5, p. 056608, 2017. (Cited on page 85.)

[222] S. Miwa, S. Ishibashi, H. Tomita, T. Nozaki, E. Tamura, K. Ando, N. Mizuochi, T. Saruya, H. Kubota, K. Yakushiji, *et al.*, "Highly sensitive nanoscale spin-torque diode," *Nature materials*, vol. 13, no. 1, p. 50, 2014. (Cited on page 85.)

[223] S. Louis, V. Tyberkevych, J. Li, I. Lisenkov, R. Khymyn, E. Bankowski, T. Meitzler, I. Krivorotov, and A. Slavin, "Low power microwave signal detection with a spin-torque nano-oscillator in the active self-oscillating regime," *IEEE Transactions on Magnetics*, vol. 53, no. 11, pp. 1–4, 2017. (Cited on page 85.)

[224] S. Hemour, Y. Zhao, C. H. P. Lorenz, D. Houssameddine, Y. Gui, C.-M. Hu, and K. Wu, "Towards low-power high-efficiency rf and microwave energy harvesting," *IEEE transactions on microwave theory and techniques*, vol. 62, no. 4, pp. 965–976, 2014. (Cited on page 85.)

[225] K. Yogendra, D. Fan, Y. Shim, M. Koo, and K. Roy, "Computing with coupled spin torque nano oscillators," in *Design Automation Conference (ASP-DAC), 2016 21st Asia and South Pacific*, pp. 312–317, IEEE, 2016. (Cited on page 86.)

[226] M. Romera, P. Talatchian, S. Tsunegi, F. A. Araujo, V. Cros, P. Bortolotti, J. Trastoy, K. Yakushiji, A. Fukushima, H. Kubota, *et al.*, "Vowel recognition with four coupled spin-torque nano-oscillators," *Nature*, p. 1, 2018. (Cited on pages 86 and 207.)

[227] H. Jaeger and H. Haas, "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication," *science*, vol. 304, no. 5667, pp. 78–80, 2004. (Cited on page 89.)

[228] J. J. Steil, "Backpropagation-decorrelation: online recurrent learning with o (n) complexity," in *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, vol. 2, pp. 843–848, IEEE, 2004. (Cited on page 89.)

[229] D. Verstraeten, B. Schrauwen, M. d'Haene, and D. Stroobandt, "An experimental unification of reservoir computing methods," *Neural networks*, vol. 20, no. 3, pp. 391–403, 2007. (Cited on pages 89 and 145.)

[230] P. Joshi and W. Maass, "Movement generation and control with generic neural microcircuits," in *International Workshop on Biologically Inspired Approaches to Advanced Information Technology*, pp. 258–273, Springer, 2004. (Cited on page 89.)

[231] H. Burgsteiner, "Training networks of biological realistic spiking neurons for real-time robot control," in *Proceedings of the 9th international conference on engineering applications of neural networks, Lille, France*, pp. 129–136, 2005. (Cited on page 89.)

[232] W. Maass, T. Natschläger, and H. Markram, "A model for real-time computation in generic neural microcircuits," in *Advances in neural information processing systems*, pp. 229–236, 2003. (Cited on page 89.)

[233] M. D. Skowronski and J. G. Harris, "Minimum mean squared error time series classification using an echo state network prediction model," in *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*, pp. 4–pp, IEEE, 2006. (Cited on page 89.)

[234] D. Verstraeten, B. Schrauwen, D. Stroobandt, and J. Van Campenhout, "Isolated word recognition with the liquid state machine: a case study," *Information Processing Letters*, vol. 95, no. 6, pp. 521–528, 2005. (Cited on pages 89 and 145.)

[235] D. Verstraeten, B. Schrauwen, and D. Stroobandt, "Reservoir-based techniques for speech recognition," in *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, pp. 1050–1053, IEEE, 2006. (Cited on page 89.)

[236] H. Jaeger, "Adaptive nonlinear system identification with echo state networks," in *Advances in neural information processing systems*, pp. 609–616, 2003. (Cited on page 89.)

[237] T. M. Cover, "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition," *IEEE transactions on electronic computers*, no. 3, pp. 326–334, 1965. (Cited on page 92.)

[238] W. Maass, P. Joshi, and E. D. Sontag, "Computational aspects of feedback in neural circuits," *PLoS computational biology*, vol. 3, no. 1, p. e165, 2007. (Cited on page 92.)

[239] L. Appeltant, M. C. Soriano, G. Van der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C. R. Mirasso, and I. Fischer, "Information processing using a single dynamical node as complex system," *Nature communications*, vol. 2, p. 468, 2011. (Cited on pages 92, 96, 101, 145, 156, and 171.)

[240] H. Jaeger, *Short term memory in echo state networks*, vol. 5. GMD-Forschungszentrum Informationstechnik, 2001. (Cited on pages 93, 174, 177, and 179.)

[241] N. Levinson, "The wiener (root mean square) error criterion in filter design and prediction," *Journal of Mathematics and Physics*, vol. 25, no. 1-4, pp. 261–278, 1946. (Cited on page 94.)

[242] R. Penrose, "A generalized inverse for matrices," in *Mathematical proceedings of the Cambridge philosophical society*, vol. 51, pp. 406–413, Cambridge University Press, 1955. (Cited on page 95.)

[243] A. N. Tikhonov, "On the solution of ill-posed problems and the method of regularization," in *Doklady Akademii Nauk*, vol. 151, pp. 501–504, Russian Academy of Sciences, 1963. (Cited on page 95.)

[244] Y. Paquot, F. Duport, A. Smerieri, J. Dambre, B. Schrauwen, M. Haelterman, and S. Massar, "Optoelectronic reservoir computing," *Scientific reports*, vol. 2, p. 287, 2012. (Cited on pages 101, 102, 113, and 145.)

[245] F. Duport, B. Schneider, A. Smerieri, M. Haelterman, and S. Massar, "All-optical reservoir computing," *Optics express*, vol. 20, no. 20, pp. 22783–22795, 2012. (Cited on pages 101, 102, and 145.)

[246] L. Larger, M. C. Soriano, D. Brunner, L. Appeltant, J. M. Gutiérrez, L. Pesquera, C. R. Mirasso, and I. Fischer, "Photonic information processing beyond turing: an optoelectronic implementation of reservoir computing," *Optics express*, vol. 20, no. 3, pp. 3241–3249, 2012. (Cited on pages 101, 102, and 145.)

[247] A. Dejonckheere, F. Duport, A. Smerieri, L. Fang, J.-L. Oudar, M. Haelterman, and S. Massar, "All-optical reservoir computer based on saturation of absorption," *Optics express*, vol. 22, no. 9, pp. 10868–10881, 2014. (Cited on pages 101, 102, and 145.)

[248] K. Vandoorne, P. Mechet, T. Van Vaerenbergh, M. Fiers, G. Morthier, D. Verstraeten, B. Schrauwen, J. Dambre, and P. Bienstman, "Experimental demonstration of reservoir computing on a silicon photonics chip," *Nature communications*, vol. 5, p. 3541, 2014. (Cited on pages 101, 102, and 145.)

[249] T. Li, K. Nakajima, M. Cianchetti, C. Laschi, and R. Pfeifer, "Behavior switching using reservoir computing for a soft robotic arm," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 4918–4924, IEEE, 2012. (Cited on page 101.)

[250] L. Larger, A. Baylón-Fuentes, R. Martinenghi, V. S. Udaltsov, Y. K. Chembo, and M. Jacquot, "High-speed photonic reservoir computing using a time-delay-based architecture: Million words per second classification," *Physical Review X*, vol. 7, no. 1, p. 011015, 2017. (Cited on pages 101, 102, and 145.)

[251] D. Brunner, M. C. Soriano, C. R. Mirasso, and I. Fischer, "Parallel photonic information processing at gigabyte per second data rates using transient states," *Nature communications*, vol. 4, p. 1364, 2013. (Cited on pages 101, 102, and 145.)

[252] Q. Vinckier, F. Duport, A. Smerieri, K. Vandoorne, P. Bienstman, M. Haelterman, and S. Massar, "High-performance pho-

tonic reservoir computer based on a coherently driven passive cavity," *Optica*, vol. 2, no. 5, pp. 438–446, 2015. (Cited on pages 101, 102, and 145.)

[253] H. Zhang, X. Feng, B. Li, Y. Wang, K. Cui, F. Liu, W. Dou, and Y. Huang, "Integrated photonic reservoir computing based on hierarchical time-multiplexing structure," *Optics Express*, vol. 22, no. 25, pp. 31356–31370, 2014. (Cited on page 101.)

[254] F. Duport, A. Smerieri, A. Akrout, M. Haelterman, and S. Massar, "Fully analogue photonic reservoir computer," *Scientific reports*, vol. 6, p. 22381, 2016. (Cited on pages 101 and 102.)

[255] D. Nikolić, S. Haeusler, W. Singer, and W. Maass, "Temporal dynamics of information content carried by neurons in the primary visual cortex," in *Advances in neural information processing systems*, pp. 1041–1048, 2007. (Cited on page 101.)

[256] F. Schürmann, K. Meier, and J. Schemmel, "Edge of chaos computation in mixed-mode vlsi-a hard liquid," in *Advances in neural information processing systems*, pp. 1201–1208, 2005. (Cited on page 101.)

[257] L. Appeltant *et al.*, *Reservoir computing based on delay-dynamical systems*. PhD thesis, Universitat de les Illes Balears, 2012. (Cited on pages 101, 102, and 156.)

[258] D. Prychynenko, M. Sitte, K. Litzius, B. Krüger, G. Bourianoff, M. Kläui, J. Sinova, and K. Everschor-Sitte, "Magnetic skyrmion as a nonlinear resistive element: A potential building block for reservoir computing," *Physical Review Applied*, vol. 9, no. 1, p. 014034, 2018. (Cited on page 103.)

[259] K. Fujii and K. Nakajima, "Harnessing disordered-ensemble quantum dynamics for machine learning," *Physical Review Applied*, vol. 8, no. 2, p. 024030, 2017. (Cited on page 103.)

[260] R. Lyon, "A computational model of filtering, detection, and compression in the cochlea," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'82.*, vol. 7, pp. 1282–1285, IEEE, 1982. (Cited on pages 145, 147, and 148.)

[261] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," in *Readings in speech recognition*, pp. 65–74, Elsevier, 1990. (Cited on pages 145 and 146.)

[262] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal processing magazine*, vol. 29, no. 6, pp. 82–97, 2012. (Cited on page 145.)

[263] S. S. Stevens, J. Volkmann, and E. B. Newman, "A scale for the measurement of the psychological magnitude pitch," *The Journal of the Acoustical Society of America*, vol. 8, no. 3, pp. 185–190, 1937. (Cited on page 146.)

[264] M. Slaney, "Auditory toolbox," *Interval Research Corporation, Tech. Rep*, vol. 10, p. 1998, 1998. (Cited on page 146.)

[265] M. Slaney, *Lyon's cochlear model*, vol. 13. Citeseer, 1988. (Cited on page 147.)

[266] G. Von Békésy, *Sensory inhibition*, vol. 4886. Princeton University Press, 2017. (Cited on page 148.)

[267] F. J. Harris, "On the use of windows for harmonic analysis with the discrete fourier transform," *Proceedings of the IEEE*, vol. 66, no. 1, pp. 51–83, 1978. (Cited on page 149.)

[268] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008. (Cited on page 154.)

[269] T. Instruments, "Ti 46-word speaker-dependent isolated word corpus (cd-rom)," *Gaithersburg: NIST*, 1991. (Cited on page 158.)

[270] B. Efron and R. Tibshirani, "Improvements on cross-validation: the 632+ bootstrap method," *Journal of the American Statistical Association*, vol. 92, no. 438, pp. 548–560, 1997. (Cited on page 166.)

[271] D. Verstraeten, J. Dambre, X. Dutoit, and B. Schrauwen, "Memory versus non-linearity in reservoirs," in *Neural Networks (IJCNN), The 2010 International Joint Conference on*, pp. 1–8, IEEE, 2010. (Cited on page 179.)

[272] J. Dambre, D. Verstraeten, B. Schrauwen, and S. Massar, "Information processing capacity of dynamical systems," *Scientific reports*, vol. 2, p. 514, 2012. (Cited on page 179.)

[273] M. Inubushi and K. Yoshimura, "Reservoir computing beyond memory-nonlinearity trade-off," *Scientific Reports*, vol. 7, no. 1, p. 10199, 2017. (Cited on page 179.)

[274] M. Gajek, J. Nowak, J. Sun, P. Trouilloud, E. O'sullivan, D. Abraham, M. Gaidis, G. Hu, S. Brown, Y. Zhu, *et al.*, "Spin torque switching of 20 nm magnetic tunnel junctions with perpendicular anisotropy," *Applied Physics Letters*, vol. 100, no. 13, p. 132408, 2012. (Cited on page 207.)

[275] B. Scellier and Y. Bengio, "Equilibrium propagation: bridging the gap between energy-based models and backpropagation," *Frontiers in computational neuroscience*, vol. 11, p. 24, 2017. (Cited on page 207.)