

UNIVERSITÉ DU QUÉBEC À MONTRÉAL

APPRENTISSAGE BIO-INSPIRÉ DANS LES RÉSEAUX DE NEURONES RÉCURRENTS

MÉMOIRE

PRÉSENTÉ

COMME EXIGENCE PARTIELLE

DE LA MAÎTRISE EN INFORMATIQUE

PAR

MATTHIEU PICARD

JUIN 2025

UNIVERSITÉ DU QUÉBEC À MONTRÉAL
Service des bibliothèques

Avertissement

La diffusion de ce mémoire se fait dans le respect des droits de son auteur, qui a signé le formulaire *Autorisation de reproduire et de diffuser un travail de recherche de cycles supérieurs* (SDU-522 – Rév.12-2023). Cette autorisation stipule que «conformément à l'article 11 du Règlement no 8 des études de cycles supérieurs, [l'auteur] concède à l'Université du Québec à Montréal une licence non exclusive d'utilisation et de publication de la totalité ou d'une partie importante de [son] travail de recherche pour des fins pédagogiques et non commerciales. Plus précisément, [l'auteur] autorise l'Université du Québec à Montréal à reproduire, diffuser, prêter, distribuer ou vendre des copies de [son] travail de recherche à des fins non commerciales sur quelque support que ce soit, y compris l'Internet. Cette licence et cette autorisation n'entraînent pas une renonciation de [la] part [de l'auteur] à [ses] droits moraux ni à [ses] droits de propriété intellectuelle. Sauf entente contraire, [l'auteur] conserve la liberté de diffuser et de commercialiser ou non ce travail dont [il] possède un exemplaire.»

REMERCIEMENTS

Je tiens à exprimer ma gratitude envers tous ceux qui ont contribué à la réalisation de cette thèse de maîtrise. Tout d'abord, je remercie sincèrement le Fonds de recherche du Québec – Nature et technologies (FQRNT) pour leur soutien financier à travers le programme de bourse qui a grandement facilité mes recherches. Je suis également reconnaissant envers le programme de stage MITACS, qui a permis d'enrichir mon parcours académique par des expériences pratiques significatives et formatrices.

Un merci spécial à mes directeurs de recherche, M. Petko Valtchev, Ph.D, et M. Alexandre Blondin-Massé, Ph.D, pour leur guidance, leur patience et leur expertise. Leur soutien et leurs précieux conseils ont été d'une aide sans mesure pour me permettre de produire ce mémoire. Leur engagement envers l'excellence m'ont permis de produire un travail qui m'est cher.

Ce travail n'aurait pas été possible sans ce soutien je suis profondément reconnaissant pour chaque contribution à mon projet de maîtrise.

TABLE DES MATIÈRES

LISTE DES FIGURES	vii
LISTE DES TABLEAUX	viii
ACRONYMES	ix
RÉSUMÉ	x
CHAPITRE 1 INTRODUCTION	1
CHAPITRE 2 SYSTÈMES DYNAMIQUES	7
2.1 Formulation.....	8
2.2 Réseaux de neurones récurrents	8
2.3 Espace des phases.....	10
2.4 Attracteurs dynamiques.....	11
2.4.1 Attracteurs sur point fixe	12
2.4.2 Cycle limite.....	12
2.4.3 Attracteur périodique discret	12
2.4.4 Attracteurs limites	14
2.5 Chaos dynamique	15
2.5.1 Attracteurs étranges.....	17
2.5.2 La limite du chaos	19
2.6 Synchronisation	20
2.6.1 Métronome	21
2.6.2 Modèle de Kuramoto.....	21
2.6.3 Un cerveau synchronisé	22
CHAPITRE 3 LE CERVEAU	23

3.1	L'hippocampe.....	25
3.1.1	Organisation des neurones dans l'hippocampe.....	25
3.2	Le cerveau comme système dynamique	26
3.2.1	Attracteur pour la navigation spatiale	26
3.2.2	Bruit stochastique	27
3.2.3	Le cerveau à la limite du chaos	27
3.3	Apprentissage en ligne en un coup	28
3.3.1	Apprentissage en ligne à long terme	29
3.4	Activation binaire.....	30
3.4.1	Réfraction	30
3.4.2	Encodage de l'intensité dans la fréquence	31
3.5	Règle d'apprentissage locale	31
3.5.1	Plasticité synaptique temporelle.....	31
3.6	Autres facteurs	33
3.6.1	Alpha, Beta, Theta, Delta	33
3.6.2	Neurotransmetteurs.....	34
3.7	Organisation des neurones dans le cortex en colonnes	34
3.7.1	Architecture modulaire.....	35
3.7.2	Dynamiques internes.....	36
3.8	Mémoire associative contextuelle.....	36
3.8.1	Intégration globale de l'information.....	36
CHAPITRE 4 ALGORITHMES		37
4.1	Propagation arrière	38

4.1.1	Algorithme original	38
4.1.2	Avancées	40
4.1.3	Limitations	42
4.2	Apprentissage Hebbien	44
4.2.1	Règle d'apprentissage locale	45
4.2.2	Apprentissage par renforcement	45
4.2.3	Temporalité	45
4.2.4	Avancées	46
4.2.5	Limites	46
4.3	Mémoires associatives (Hopfield)	46
4.3.1	Algorithme original	47
4.3.2	Avancées	47
4.3.3	Limitations	49
4.4	Apprentissage compétitif	50
4.4.1	Cartes de Kohonen	50
4.5	Attracteurs spatio-temporel hebbien	52
4.5.1	Algorithme	52
CHAPITRE 5	CONTRIBUTION	55
5.1	Méthodologie	56
5.1.1	Cadre expérimental	57
5.1.2	Contraintes biologiques	59
5.1.3	Attracteurs limites	61
5.1.4	Inspiration algorithmique	62

5.1.5	Notation.....	64
5.1.6	Architecture.....	66
5.2	Proposition	66
5.2.1	Activation	66
5.2.2	Paramètres globaux	73
5.3	Résultats	73
5.3.1	Analyse visuelle.....	76
5.4	Optimisations.....	77
5.4.1	Cycles forcés	79
5.4.2	Détermination initial du signe des poids	79
5.4.3	Affaiblissement temporel des poids	79
CHAPITRE 6	CONCLUSION	81
BIBLIOGRAPHIE	85

LISTE DES FIGURES

Figure 2.1	Un réseau de neurone récurrent basique à une couche de 4 neurones	9
Figure 2.2	Un réseau de neurone récurrent simple avec entrée/sortie à une couche cachée	10
Figure 2.3	Espace des phases d'un système représentant le mouvement d'un pendule simple avec et sans coefficient de friction.	13
Figure 2.4	Un attracteur périodique discret dans un réseau récurrent simple.	14
Figure 2.5	Un attracteur limite dans l'espace des phases d'un réseau récurrent simple.....	16
Figure 2.6	Attracteur de Lorenz	18
Figure 2.7	Valeurs des états de différents neurones évoluant à travers le temps dans un réseau de neurones récurrents simple. On voit ici qu'il existe un attracteur limite dans un système en raison du motif périodique visible dans ce graphique. L'axe vertical est la valeur du neurone à un temps donné et l'axe horizontal est le nombre d'itérations (ou "step" en anglais) écoulées ...	20
Figure 2.8	Visualisation de l'espace des phases de métronomes (ou oscillateurs) légèrement couplés dans un système simulé.....	21
Figure 5.1	Évolution de l'apprentissage pour différentes tailles de RN	58
Figure 5.2	Évolution de l'apprentissage de notre algorithme bio-RNN	74
Figure 5.3	Les matrices de $W_{0,1}$ pour un ensemble de neurones après l'apprentissage	76
Figure 5.4	Les matrices de $W_{1,1}$ pour un ensemble de neurones après l'apprentissage	78
Figure 5.5	Les matrices de $W_{0,1}$ pour un ensemble de neurones après l'apprentissage, avec un affaiblissement temporel des poids	80

LISTE DES TABLEAUX

Tableau 5.1	Paramètres	73
-------------	------------------	----

ACRONYMS

UQAM Université du Québec à Montréal.

IA Intelligence Artificielle.

AGI Intelligence Artificielle Générale.

RP Rétropropagation.

RN Réseau de neurones.

RNP Réseau de neurones profond.

EEG Électro-encéphalogramme.

LSTM Réseau de neurone à longue mémoire à court terme, de l'anglais "Long short-term memory".

STDP Spike-Time Dependent Plasticity.

CNN Réseau de neurones convolutif, de l'anglais "Convolutionnal Neural Network".

RNN Réseau de neurone récurrent.

bioRNN Réseau de neurone récurrent bio-inspiré ou bio-plausible.

LLM Modèle de langage massif, de l'anglais "Large Language Model".

RÉSUMÉ

Ce mémoire se penche sur la synergie entre divers paradigmes d'apprentissage biologique et algorithmique. En mettant l'accent sur la complémentarité et le potentiel de divers mécanismes d'apprentissage biologiques à être formalisés mathématiquement dans un modèle, nous cherchons à éclairer leur capacité à offrir des alternatives à l'algorithme de la rétropropagation (RP) dans les réseaux de neurones profonds (RN). Pour ce faire, nous explorons l'intersection entre ces paradigmes, en mettant l'accent sur les mécanismes d'apprentissage Hebbien et anti-Hebbien et la plasticité synaptique à temps dépendant (STDP), pour étudier les dynamiques internes d'un système de neurones récurrents les intégrant. Notre objectif est d'établir s'il est possible de créer un système artificiel étant au moins partiellement analogue à sa contrepartie biologique dans son fonctionnement dynamique et son apprentissage (entre deux neurones) présentant des propriétés d'apprentissage (globales).

Tout en reconnaissant l'efficacité de la rétropropagation, qui a permis des avancées significatives en intelligence artificielle (IA) pour la génération et la classification de données multimodales, pour le traitement du langage naturel et pour l'apprentissage par renforcement, cette recherche souligne ses limites qui sont l'objet de nombreuse recherche de pointe.

Notre travail adopte une approche à la fois théorique et expérimentale afin d'évaluer comment ces principes, plus biologiquement plausibles et inspirés du fonctionnement du cerveau et de sa topologie dynamique que les méthodes courantes, peuvent offrir des processus d'apprentissage plus rapides et adaptables. Pour le volet expérimental, nous proposons un algorithme d'apprentissage bio-inspiré pour les réseaux de neurones récurrents (bio-RNN). Nos résultats suite à son implémentation montrent que le modèle proposé est initialement performant et rapide dans l'apprentissage, suggérant une piste intéressante de recherche. Cependant, cet avantage disparaît rapidement lorsqu'on alloue un temps d'entraînement raisonnable et notre méthode demeure bien en deçà des performances de la rétropropagation en termes de précision et de rappel absolus sur des tâches complexes. Nous sommes néanmoins parvenus, à travers nos expériences, à démontrer que l'intégration des principes Hebbiens dans les architectures neuronales récurrentes dynamiques est faisable et prometteuse, soulignant la nécessité de poursuivre nos recherches pour mieux comprendre et affiner ces méthodes alternatives.

CHAPITRE 1
INTRODUCTION

Grâce à l'avènement des modèles de réseaux de neurones (RN), l'intelligence artificielle (IA) a fait des avancées considérables, s'imposant comme une discipline centrale pour la compréhension et la modélisation de données massives et complexes. Que ce soit en géologie (Huang *et al.*, 2017), en physique (Thurey *et al.*, 2022), en biologie (Skolnick *et al.*, 2017) ou même en chimie (Mater et Coote, 2019), il est désormais difficile de trouver un champ de recherche au sein duquel cette technique ne s'est point infiltrée. L'apprentissage profond (AP) est maintenant appliqué pour le traitement des données à travers les sciences naturelles.

La neuroscience est l'un de ces champs qui ont été propulsés vers l'avant grâce à ces techniques. D'un côté, les chercheurs en neuroscience utilisent désormais largement des modèles de pointe afin d'interpréter les données complexes de télémétrie qu'ils récoltent. Que ce soit pour l'imagerie à résonance magnétique (Shahamat et Saniee Abadeh, 2020; Zhang *et al.*, 2023), les flux en provenance d'implants (Zhang *et al.*, 2019), les images microscopiques (Xing *et al.*, 2018) ou les données électro-encéphalographiques (EEG) (Gao *et al.*, 2021; Schirrneister *et al.*, 2017), l'apprentissage profond a fourni un nouvel outil d'analyse repoussant les frontières de notre connaissance. De l'autre, la création de modélisations visant différents processus neurologiques (Wixted *et al.*, 2014; Wert-Carvajal *et al.*, 2022), cognitifs (Zhang *et al.*, 2023) et computationnels (Wixted *et al.*, 2014; Zhang *et al.*, 2023; Tomasello *et al.*, 2018) soutenant l'avènement de la conscience leur permet de jeter une nouvelle lumière sur le fonctionnement interne du cerveau.

Pour ce qui est des algorithmes supportant ces modèles, les chercheurs en IA ont fait des progrès phénoménaux. Il suffit de penser aux modèles massifs de langage (LLM) dont les avancées récentes nous amènent à reconsidérer la notion même d'intelligence. Parmi les techniques ayant joué un rôle pivot dans cette révolution, la propagation arrière, ou rétropropagation (RP), se distingue comme le mécanisme d'apprentissage fondamental dans les réseaux de neurones profonds (RNP) à travers une forte majorité de leurs applications. Que ce soit dans les modèles de diffusion (Ho *et al.*, 2020) ou les réseaux génératifs adversariaux pour la génération d'image (Goodfellow *et al.*, 2020), les LSTM (Hochreiter et Schmidhuber, 1997) ou les transformeurs (Vaswani *et al.*, 2017) pour le traitement du langage naturel ou encore dans les techniques d'apprentissage par renforcement comme le "Q-Learning" (Mnih *et al.*, 2013) et le système AlphaZero (Silver *et al.*, 2017), on retrouve toujours la rétropropagation derrière leur entraînement et le progrès fulgurant des dernières années n'aurait pas été possible sans elle.

La RP est un algorithme capable de résoudre un grand nombre de problèmes d'optimisation qui étaient jusqu'alors intraitables et de surclasser marginalement un grand nombre de techniques modernes sur de

nombreux autres. Il est donc naturel qu'elle se soit imposée comme l'algorithme de prédilection pour une panoplie d'applications de résolution de problèmes dans différents domaines de recherche et dans l'industrie. C'est pourquoi il est généralement accepté que les réseaux de neurones profonds et la propagation arrière sont indispensables pour faire la classification d'image, la génération d'image et pour le traitement du langage naturel à des niveaux atteignant l'état de l'art.

La propagation arrière, lorsqu'appliquée à l'entraînement des réseaux de neurones profonds, repose sur un principe d'optimisation mathématique pour ajuster les poids synaptiques dudit réseau en fonction de l'erreur de sortie. Son efficacité à traiter des problèmes complexes et sa contribution au succès des applications d'IA modernes sont indéniables. Cependant, cela ne l'empêche pas de présenter certaines limites, notamment sa tendance à nécessiter une importante quantité de données et de larges ressources computationnelles, son apprentissage lent et progressif, et l'oubli de certains apprentissages initiaux au cours de l'entraînement (Kemker *et al.*, 2018). La question sur sa capacité à être l'algorithme soutenant l'intelligence artificielle générale (tiré de l'anglais, Artificial General Intelligence, AGI dans le texte) demeure à ce jour ouverte, puisque la rétropropagation n'est pas bio-plausible, ce qui veut dire que nous n'avons toujours pas découvert de mécanismes analogues dans le cerveau biologique, qui est le seul système connu étant capable d'atteindre l'intelligence générale.

Parallèlement, l'apprentissage Hebbien (Hebb, 1949), inspiré par fonctionnement du cerveau, offre une perspective différente et complémentaire sur la façon dont les systèmes intelligents peuvent apprendre et s'adapter. Celui-ci, évoquant le célèbre adage "les neurones qui s'activent ensemble se lient ensemble", s'inscrit dans une démarche visant à imiter les processus d'apprentissage naturels observés dans le cerveau. Bien que souvent considéré comme inférieur à la rétropropagation (Krotov et Hopfield, 2019), l'apprentissage Hebbien, se rapproche davantage des phénomènes d'apprentissage biologique, suggérant une voie vers une capacité d'apprentissage universelle et rapide semblable à celle du cerveau humain.

Cette approche, bien que moins répandue dans les applications d'IA contemporaines, a été historiquement l'objet de nombreuses recherches dans la littérature. Avec le succès de la BP, le rythme de publications sur les algorithmes bio-inspirés et bio-plausibles s'est quelque peu estompé au cours de la dernière décennie, mais semble cependant connaître un regain au cours des dernières années ¹.

¹ Il s'agit ici d'une observation empirique, basée sur notre revue de littérature

Avec la croissance constante des modèles de pointe, les limitations de la BP deviennent de plus en plus apparentes. Leur taille colossale nécessitant des ressources computationnelles massives et des jeux de données massifs pour leur entraînement les rend inaccessibles pour beaucoup de chercheurs n'ayant pas accès à ces ressources. Les différentes techniques Hebbiennes présentant l'avantage d'une plus grande bio-plausibilité, soulèvent la possibilité d'atteindre une forme d'apprentissage plus rapide, performante, adaptable et permettant de mieux traiter les flux de données. Le tout en augmentant l'accessibilité et le potentiel de démocratisation de la technologie s'il est possible d'augmenter d'ordres de magnitudes la vitesse de convergence tant au niveau de la demande en calcul que de la demande en données d'entraînement. De plus, l'état plus embryonnaire du domaine fait en sorte que des gains considérables peuvent être effectués avec des moyens relativement modestes.

Avec ce mémoire, nous proposons une exploration de la synergie entre différents paradigmes d'apprentissage biologiques et algorithmiques, en nous interrogeant sur leur complémentarité et leur potentiel à enrichir notre compréhension de l'intelligence artificielle. À travers une revue critique de la littérature et le développement de modèles et d'expériences numériques, nous chercherons à déterminer dans quelle mesure la combinaison de différents processus modélisés mathématiquement peut ouvrir de nouvelles perspectives pour la conception de systèmes intelligents plus performants et plus proches de la fonction cognitive humaine. Par conséquent, ce travail s'inscrit dans une démarche à la fois théorique et expérimentale, visant à contribuer au dialogue entre l'apprentissage machine (de l'anglais Machine Learning, ou ML) et les neurosciences, dans l'espoir de rapprocher les machines des capacités et de la flexibilité des systèmes d'apprentissage biologiques.

En s'attaquant à ce sujet, ce mémoire vise à offrir une perspective interdisciplinaire entre l'intelligence artificielle et les neurosciences, dans le but d'investiguer des pistes de recherche pour la conception d'une nouvelle génération d'algorithmes d'apprentissage qui combinent l'efficacité mathématique de la propagation arrière avec la flexibilité et la robustesse de l'apprentissage Hebbien. Ce travail se positionne à l'intersection de la recherche fondamentale et appliquée, avec l'espoir de jeter des ponts entre les modèles théoriques d'apprentissage et leur mise en œuvre pratique dans des systèmes d'IA avancés. L'interaction entre les différents paradigmes d'apprentissages bio-plausibles, ainsi que leur application à des architectures neuronales dynamiques récurrentes, constitue le cœur de cette recherche.

Afin d'explorer ces thèmes variés pour ensuite en extraire les éléments saillants sous forme d'un algorithme, ce mémoire est divisé en quatre axes principaux: les systèmes dynamiques, l'exploration des avancées

théoriques et expérimentales en neurosciences, l'étude approfondie des différents algorithmes d'apprentissage pour les réseaux de neurones standard et récurrent, et enfin, notre proposition d'un algorithme d'apprentissage dynamique semi-supervisé visant à valider l'importance des différentes contraintes et méthodes abordées.

Le chapitre 2 porte donc sur les systèmes dynamiques chaotiques et vise à établir une base théorique commune pour la compréhension des mécanismes dirigeant les dynamiques dans notre modèle et de l'analyse de nos résultats.

Pour le troisième chapitre, nous nous attardons à établir les fondations biologiques sur lesquelles repose notre compréhension de l'apprentissage et de la mémoire. En mettant en lumière les mécanismes neuronaux, tels que la plasticité synaptique et les dynamiques neuronales complexes, ce chapitre vise à établir une base théorique derrière les processus cognitifs biologiques et leur modélisation informatique. En explorant les systèmes dynamiques et les motifs d'activation neuronaux, nous soulignons l'importance de ces phénomènes dans la formation de la mémoire et dans les processus d'apprentissage, établissant ainsi un cadre pour l'intégration de ces principes dans notre proposition.

Le chapitre 4 porte quant à lui sur divers algorithmes d'apprentissage utilisés pour optimiser différents types de réseaux de neurones. Nous commençons par présenter la technique la plus utilisée, soit la rétropropagation, et propose une analyse critique de ses contraintes intrinsèques, telles que sa demande computationnelle élevée et son manque de bio-plausibilité. Il établira également le contexte motivant la recherche d'approches alternatives ou complémentaires en IA. Ensuite, nous nous penchons sur l'apprentissage Hebbien sous différentes formes et leur potentiel à simuler plus fidèlement les mécanismes d'apprentissage biologiques, avec un accent particulier sur leur application au sein de réseaux neuronaux récurrents dynamiques. Ce chapitre explore la capacité de ces méthodes à offrir des voies d'apprentissage plus naturelles et efficaces, en particulier pour le traitement de données temporelles et séquentielles.

Subséquentement, au chapitre 5, nous formalisons la proposition de notre algorithme. Notre approche vise à fusionner les différents concepts qui sont présentés au sein d'un cadre unifié. Cette synthèse ambitionne de créer une fonction d'apprentissage qui, non seulement pallie les faiblesses de chaque approche prisent isolément, mais bénéficie également de leur complémentarité pour améliorer la performance globale de notre modèle. Pour ce faire, nous proposons notre architecture avec une nouvelle fonction de mise à jour des états et d'apprentissage et la comparons à une architecture simple de ANN avec BP. Pour ce faire, nous

présentons d'abord ses différentes composantes algorithmiques ainsi que différentes avenues que nous avons explorées lors de notre implémentation en Python. Cette partie se termine par une présentation des résultats des expériences effectuées dans le cadre de notre recherche.

Finalement, la conclusion de ce mémoire propose une synthèse des découvertes réalisées à travers ces axes de recherche, démontrant l'interaction complexe entre les principes neurologiques et leur application dans le domaine de l'intelligence artificielle. Nous nous attardons également sur les limites de notre approche et proposons différentes pistes d'améliorations candidates pour des recherches ultérieures plus approfondies.

CHAPITRE 2

SYSTÈMES DYNAMIQUES

Depuis la publication séminale de Lorenz (Lorenz, 1972) sur les attracteurs dynamiques qui a fait l'introduction pour la première fois de "l'effet papillon", ou sensibilité aux conditions initiales, l'étude du chaos est devenue un champ de recherche hautement interdisciplinaire qui a des applications nombreuses et variées. Que ce soit pour la météorologie comme dans l'article original, ou pour l'étude de l'évolution de populations, de la dynamique des fluides ou du mouvement des groupements des corps célestes en astrophysique, cette approche analytique nous a permis d'interpréter de nombreux phénomènes complexes qui semblaient complètement aléatoires jusque là.

Ce chapitre vise à présenter brièvement la notion de chaos et d'attracteurs dans un système dynamique et à détailler les types d'attracteurs principaux qui sont pertinents pour notre recherche.

2.1 Formulation

Formellement, un système dynamique est un système S qui comprend une fonction de la forme $f(x)$ qui détermine la dépendance temporelle d'un point dans l'espace des états possibles, autrement appelé l'espace des phases. Plus simplement, on peut l'exprimer comme:

$$\dot{x} = f(x) \tag{2.1}$$

Où $x \in \mathbb{R}^n$ et $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ sont une fonction lisse, dans le cadre du présent travail, le type de système dynamique qui nous intéresse plus spécifiquement est celui des réseaux de neurones récurrents (RNN).

2.2 Réseaux de neurones récurrents

Les systèmes dynamiques peuvent prendre de nombreuses formes. La version sur laquelle nous nous pencherons plus particulièrement pour notre recherche est celle des RNN. Ceux-ci peuvent être représentés comme un graphe composé de noeuds, ou neurones, et d'arêtes, aussi appelés poids, connexions ou synapses, qui lient ces différents noeuds entre eux. Chacun des noeuds dans l'ensemble X ont une valeur qui représente leur état, avec x_i le neurone à la position i .

Les connexions sont représentées sous la forme d'une matrice des poids W où $w_{i,j}$ est la connexion entre les neurones x_i et x_j . À chaque itération t du système, on le met à jour avec:

$$X_{t+1} = \phi(X_t W) \quad (2.2)$$

Où ϕ est une fonction d'activation arbitraire, par exemple la tangente hyperbolique réciproque $\tanh(x)$, la fonction logistique $\sigma(x) = \frac{1}{1 + e^{-x}}$, une fonction binaire $\text{bin}(x) \in \{0, 1\}$ ou de signe $\text{sign}(x) \in \{-1, 1\}$.

Définition 2.1 (Motif) Un motif $Y = \{y_0, y_1, \dots, y_n\}$ dans un RNN est une configuration donnée de neurones activés simultanément à un temps t d'un sous-ensemble de neurones de façon à ce que $Y \subset X$.

L'architecture de RNN à une couche illustrée dans 2.1 est la forme la plus simple de ce type de réseau. Les réseaux de Hopfield dont il sera question au chapitre 4 sont de ce type.

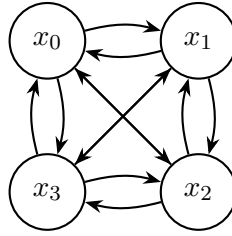


Figure 2.1 Un réseau de neurone récurrent basique à une couche de 4 neurones

Une autre architecture courante est le RNN multicouche avec couche d'entrée. Cela nous permet de séparer l'entrée du traitement de l'information (par les couches cachées) dans différentes matrices de poids afin de créer une représentation abstraite pouvant être optimisée afin de générer le motif d'activation désiré. Cette architecture est composée d'une couche d'entrée et d'une couche cachée. Les connexions W vont de la couche précédente à la suivante et à elle-même dans le cas des couches cachées. Le réseau est mis à jour avec:

$$\begin{aligned} X_{t+1}^0 &= \phi(X) \quad (\text{couche d'entrée}) \\ X_{t+1}^l &= \phi(W^{l-1,l} X^{l-1} + W^{l,l} X_t^l) \quad (\text{couche(s) cachée(s)}) \end{aligned} \quad (2.3)$$

Cependant, l'absence de couche de sortie dans ces deux types de RNN limite leur utilisation puisqu'il n'est pas possible d'effectuer un traitement dirigé de l'information (d'associer une entrée A avec une sortie désirée B telle que $B = f(A)$) tout en permettant au réseau d'apprendre un motif optimisé (à l'entraînement)

en réponse à un stimuli intrant. Les réseaux dynamiques récurrents proposés par Molter et al. dans (Molter et al., 2004; Molter et al., 2005) présentés au chapitre 4 sont un exemple de ce type de RNN.

Pour palier à ce problème, il suffit d'ajouter une couche de sortie dénuée de connexions récurrentes connectée à la dernière couche récurrente de sorte que:

$$\begin{aligned}
 X_{t+1}^0 &= \phi(X) \quad (\text{couche d'entrée}) \\
 X_{t+1}^l &= \phi(W^{l-1,l} X_t^{l-1} + W^{l,l} X_t^l) \quad (\text{couche(s) cachée(s)}) \\
 X_{t+1}^L &= \phi(W^{L-1,L} X_t^{L-1}) \quad (\text{couche de sortie})
 \end{aligned} \tag{2.4}$$

Cet ajout en fait un RNN multicouche avec entrée/sortie, tel qu'illustré par 2.2. Sous cette configuration, l'entrée, le traitement de l'information et l'inférence de la sortie sont segmentés afin de permettre au RNN d'apprendre la représentation interne (motifs d'activation des couches cachées) la plus optimale lui permettant de générer la sortie désirée.

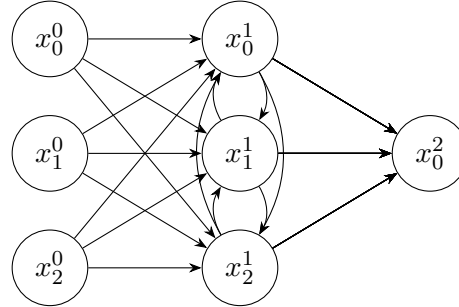


Figure 2.2 Un réseau de neurone récurrent simple avec entrée/sortie à une couche cachée

Le modèle que nous proposons au chapitre 5 est de cette forme.

2.3 Espace des phases

Dans l'étude des systèmes dynamiques, l'espace des phases est une représentation géométrique, où chaque point dans cet espace représente un état possible du système. Pour un système avec n variables, l'espace des phases est n -dimensionnel, où chaque axe représente une des variables du système. Les trajectoires dans cet espace décrivent comment les états du système évoluent dans le temps. Cet espace apporte une lumière essentielle pour visualiser et comprendre la dynamique du système, y compris les attracteurs, les

cycles limites, et les comportements chaotiques qui sont présentés plus bas. Par exemple, dans un système mécanique simple comme un pendule, l'espace des phases peut être utilisé pour montrer les positions et les vitesses du pendule à chaque instant, offrant une vue complète de son mouvement au fil du temps. La figure 2.3 illustre un tel espace des états.

2.4 Attracteurs dynamiques

Un concept clé dans l'étude et la compréhension des systèmes dynamiques est celui d'un attracteur. Formellement, un attracteur dans un système dynamique est un ensemble d'états (points dans l'espace de phase), invariant sous un régime dynamique prédéfini, vers lesquels les états voisins s'approchent asymptotiquement au fil du temps.

Définition 2.2 (Attracteur dynamique) Prenons un système dynamique décrit par l'équation $\dot{x} = f(x)$. Un attracteur est un ensemble $A = \{a_0, a_1, \dots, a_m\} \subset \mathbb{R}^n$ tel que pour tout $x \in A$ et $t > 0$,

$$\phi_t(x) \in A,$$

où ϕ_t est le flux du système, pour lequel il existe un ensemble ouvert U (le bassin d'attraction) contenant A tel que pour chaque $x \in U$,

$$\lim_{t \rightarrow \infty} \phi_t(x) \in A.$$

Dans le contexte de notre recherche, les attracteurs dans les réseaux neuronaux, particulièrement dans des modèles comme les réseaux de Hopfield qui seront présentés au chapitre 4, peuvent être vus comme des états stables ou des états de mémoire dans lesquels le réseau se stabilise, après avoir commencé à partir de certaines conditions initiales. Dans le paysage énergétique d'un tel réseau, chaque attracteur correspond à un minimum local, et le processus d'évolution dynamique dans le réseau est semblable à l'état du système descendant dans le paysage énergétique jusqu'à ce qu'il se stabilise dans l'un de ces minima (Hopfield, 1982).

Afin d'établir un langage commun pour la suite, il est important de définir les différents types d'attracteurs principaux qui apparaissent dans la littérature et dans la présentation de nos résultats.

2.4.1 Attracteurs sur point fixe

Les attracteurs sur un point fixe sont des états stables d'un système dynamique où, indépendamment des conditions initiales dans un certain voisinage, le système évolue vers et reste dans cet état stable.

Définition 2.3 (Attracteur sur point fixe) *Formellement un attracteur dynamique A est dit sur un point fixe si $|A| = 1$. Simplement, c'est un attracteur qui ne contient qu'un seul point dans l'espace de phases.*

Ces attracteurs sont des solutions d'équilibre, où le système n'exprime aucun changement au fil du temps une fois l'état atteint. Un exemple classique d'attracteur sur un point fixe est le système d'un pendule (avec friction) à l'arrêt, où, indépendamment de la position initiale du pendule, il revient toujours à sa position stable verticale et y reste.

Dans la figure 2.3, pour le système avec un coefficient de friction (en orange), le point (0,0) est un attracteur sur point fixe, qui représente l'état du pendule à l'arrêt. On peut voir que la trajectoire tend ultimement vers ce point alors que $\lim_{t \rightarrow \infty}$.

2.4.2 Cycle limite

Un cycle limite est une trajectoire fermée dans l'espace des phases d'un système dynamique, caractérisant un comportement oscillatoire stable et périodique. Un exemple concret de cycle limite peut être observé dans les oscillations de la population de certaines espèces dans un écosystème, comme le montrent les équations de Lotka-Volterra pour les modèles prédateur-proie. (Volterra, 1926)]

2.4.3 Attracteur périodique discret

Les attracteurs périodiques discrets décrivent des systèmes dynamiques qui, après une période initiale de convergence, oscillent périodiquement entre un nombre fini d'états formant un cycle répétitif. Ces cycles représentent des processus périodiques ou des oscillations dans le système où, après un certain temps,

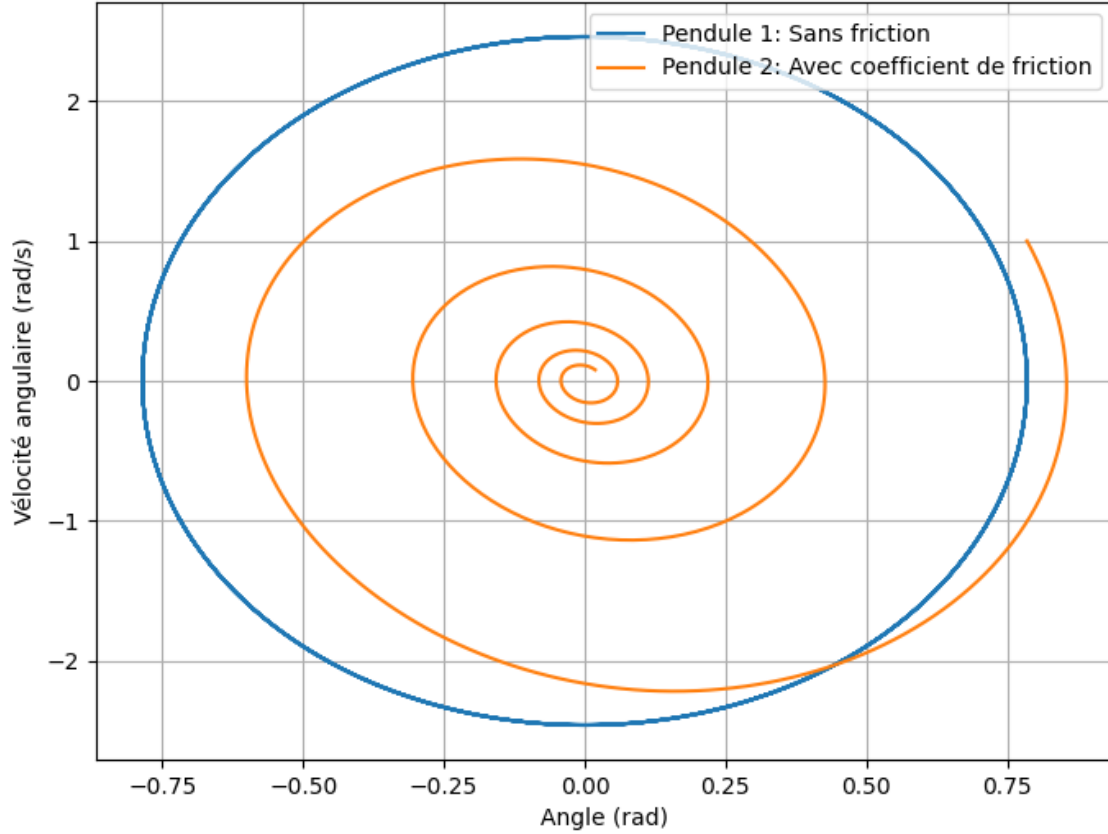


Figure 2.3 Espace des phases d'un système représentant le mouvement d'un pendule simple avec et sans coefficient de friction.

les valeurs des variables du système reviennent à leurs états précédents, répétant une série de motifs indéfiniment.

Définition 2.4 (Attracteur périodique discret) *Un attracteur périodique discret A est un attracteur pour lequel $1 < |A| < \infty$. Simplement, c'est un attracteur qui est composé d'un nombre fini de points dans l'espace de phases.*

La figure 2.4 est une illustration d'un tel attracteur dans un RNN simple composé de 3 neurones pleinement connectés entre eux. La matrice des poids permettant d'obtenir cet attracteur est

$$\begin{bmatrix} -1 & 1 & 1 \\ -1 & 1 & 1 \\ -1.1939 & -0.6108 & -0.3535 \end{bmatrix}$$

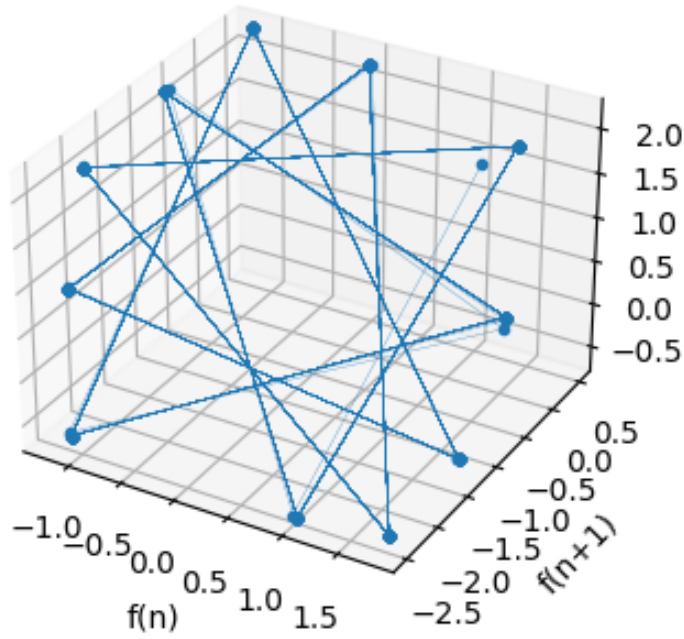


Figure 2.4 Un attracteur périodique discret dans un réseau récurrent simple.

avec un biais de $[0.2805, 0.7711, -0.9976]$ et une fonction d'activation non continue appliqués respectivement à chaque neurone à chaque itération.

2.4.4 Attracteurs limites

La notion d'attracteur limite se réfère à des comportements périodiques dans des systèmes dynamiques où les trajectoires convergent vers une orbite fermée. Contrairement aux attracteurs fixes, qui tombent sur un point, ou les attracteurs périodiques discrets, qui tendent sur un nombre fini de points, les attracteurs limites sont des ensembles infinis de points formant des chemins fermés, des trajectoires, démontrant des dynamiques continues et périodiques dans le temps. Formellement, un attracteur limite est simplement un cycle limite stable.

Définition 2.5 (Cycle limite) *Un cycle limite est un ensemble fermé $\Gamma \subset \mathbb{R}^n$ qui est une solution périodique de l'équation différentielle 2.1. Formellement, cela signifie que pour un point $x_0 \in \Gamma$ et une période $T > 0$, nous avons:*

$$\phi_T(x_0) = x_0,$$

où ϕ_t est le flot généré par \dot{x} , et pour tout $t \in \mathbb{R}$,

$$\phi_t(x_0) \in \Gamma.$$

Le cycle limite est dit "stable" si, pour tout point y dans un voisinage U de Γ , la trajectoire $\phi_t(y)$ converge vers Γ lorsque $t \rightarrow \infty$. Mathématiquement, cela peut être exprimé comme:

$$\lim_{t \rightarrow \infty} \text{dist}(\phi_t(y), \Gamma) = 0, \quad \forall y \in U.$$

Le cycle limite est dit "instable" si les trajectoires dans un voisinage de Γ s'en éloignent au fil du temps.

Le battement du cœur humain peut être considéré comme un attracteur limite, où le système cardiovasculaire oscille de manière périodique pour pomper le sang, illustrant une orbite fermée d'activités qui se répètent dans un cycle de vie stable malgré les variations entre les différents battements. Un cycle de prédateur-proie dans un écosystème, où les populations de prédateurs et de proies oscillent de manière périodique en est un autre exemple. Chaque espèce influence cycliquement l'autre, entraînant un motif répétitif et prévisible de croissance et de déclin qui ne se dissipe pas avec le temps.

Le système dont l'espace des phases (pour un neurone) est illustré dans la figure 2.5 est un simple réseau de neurones récurrents avec 2 neurones pleinement connectés entre eux. La matrice des poids permettant d'obtenir cet attracteur est $\begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}$. Une fonction d'activation $\tanh(x)$ est appliquée à la valeur de l'état d'un neurone pour la mettre à jour et un biais de $[0.4, 0.0]$ est ajouté respectivement à chaque neurone à chaque itération.

2.5 Chaos dynamique

De nombreux phénomènes, la turbulence en est un exemple, semblent à prime abord exhiber un comportement aléatoire, bien que basé sur des règles déterministes. Dans ces systèmes la moindre différence infinitésimale entre deux systèmes autrement similaires peut mener à des résultats hautement différenciés. Si nous prenons l'exemple de la météorologie, malgré notre compréhension des règles qui la gouverne, nous sommes

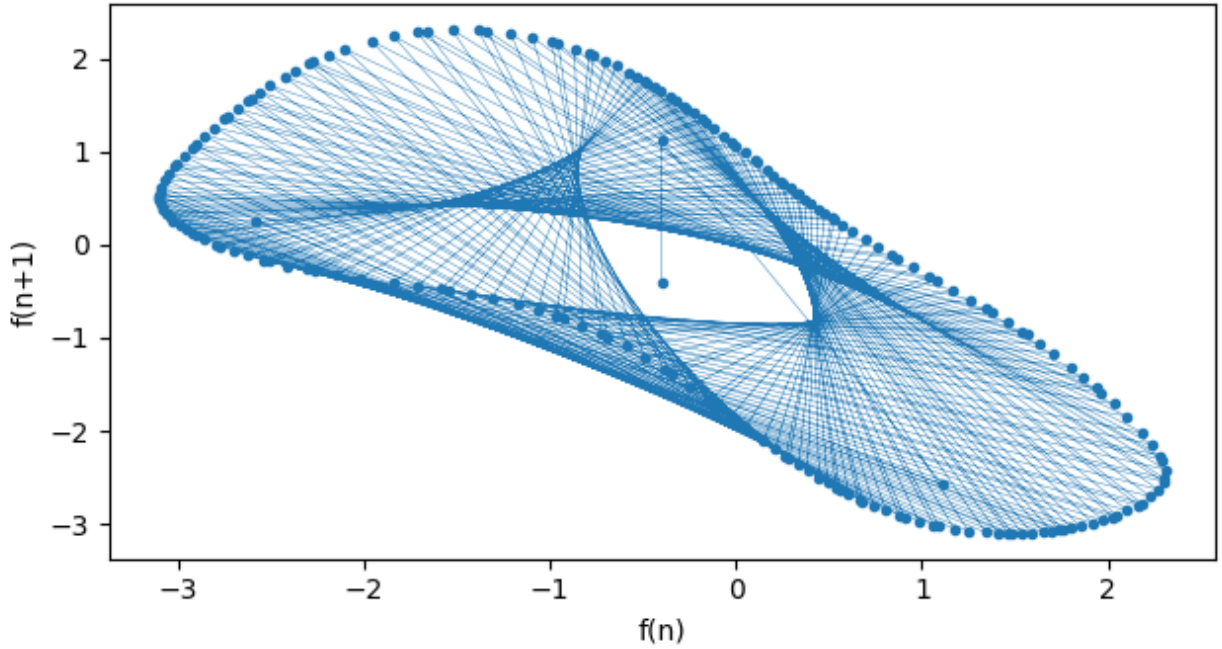


Figure 2.5 Un attracteur limite dans l'espace des phases d'un réseau récurrent simple

incapables de créer des modèles qui parviennent à prédire le système passé un certain horizon temporel puisque la connaissance parfaite de l'état du système (par exemple à chaque atome près) n'est pas possible. Même si elle l'était, le problème computationnel de prédire les états successifs serait intraitable de par la capacité massive de calcul qu'une telle simulation requerrait. En physique, le problème des trois corps (Gutzwiller, 1998) ou celui du pendule double (Hesse *et al.*, 2018) est également un exemple où, ne pouvant obtenir une mesure parfaite de la position des objets du système, nous atteignons toujours un horizon de prédiction infranchissable. On dit ce type de système sensible aux conditions initiales.

Définition 2.6 Sensibilité aux conditions initiales : Un système dynamique est dit sensible aux conditions initiales s'il existe une constante $\delta > 0$ telle que, pour toute paire de conditions initiales distinctes (x_0, y_0) , aussi proches que l'on veut, la distance entre leurs trajectoires respectives $d(x_t, y_t)$ dans l'espace des états devient supérieure à δ après un certain temps t . Cela signifie que de petites perturbations dans les conditions initiales peuvent entraîner des divergences exponentielles dans l'évolution du système, rendant ainsi la prédiction à long terme impossible.

Lorenz, météorologue qui a fait la découverte de ce type de système lors de simulations météorologiques informatiques et dont la publication séminale (Lorenz, 1963) a ouvert ce nouveau champ d'étude, définit le

chaos simplement comme (Danforth, 2013):

Définition 2.7 (Chaos) *Lorsque le présent prédit le futur, mais une approximation du présent ne détermine pas approximativement le futur¹*

cette approche théorique est pertinente dans le cadre de notre recherche, car le cerveau et l'IA ne font pas exception aux nombreux domaines des sciences naturelles où ce champ théorique peut être appliqué. de nombreux outils développés pour l'étude de systèmes chaotiques sont désormais utilisés en neuroscience (Bob, 2007; Kargarnovin et al., 2023; Justin et al., 2019; Korn et Faure, 2003). sinon, les modèles proposés par Molter sont des exemples d'application de cette théorie pour la recherche en IA (Molter et al., 2006; Molter et al., 2007; Molter et al., 2004; Molter et al., 2005) .

Malgré l'apparence aléatoire de l'évolution de ces systèmes, lorsque nous tournons notre regard vers l'espace des phases de ceux-ci, différentes structures émergent, permettant de les analyser sous un nouveau jour. La figure 2.6 est un exemple de ce type d'attracteur étrange.

2.5.1 Attracteurs étranges

On retrouve les attracteurs étranges dans des systèmes dynamiques chaotiques présentant une sensibilité extrême aux conditions initiales, où les trajectoires semblent être tirées de manière aléatoire mais sont en fait déterministes, ayant une structure fractale (une dimension de Hausdroff non entière) et confinées à une région de l'espace des phases.

Définition 2.8 (Attracteur étrange) *Un attracteur A est dit étrange s'il rencontre les propriétés suivantes:*

1. **Invariant:** *Si $x \in A$, alors la trajectoire $\phi_t(x)$ reste dans A pour tout $t \geq 0$.*
2. **Attraction :** *Il existe un voisinage U autour de A tel que pour tout $x \in U$,*

$$\lim_{t \rightarrow \infty} \text{dist}(\phi_t(x), A) = 0.$$

¹ traduction libre de l'anglais

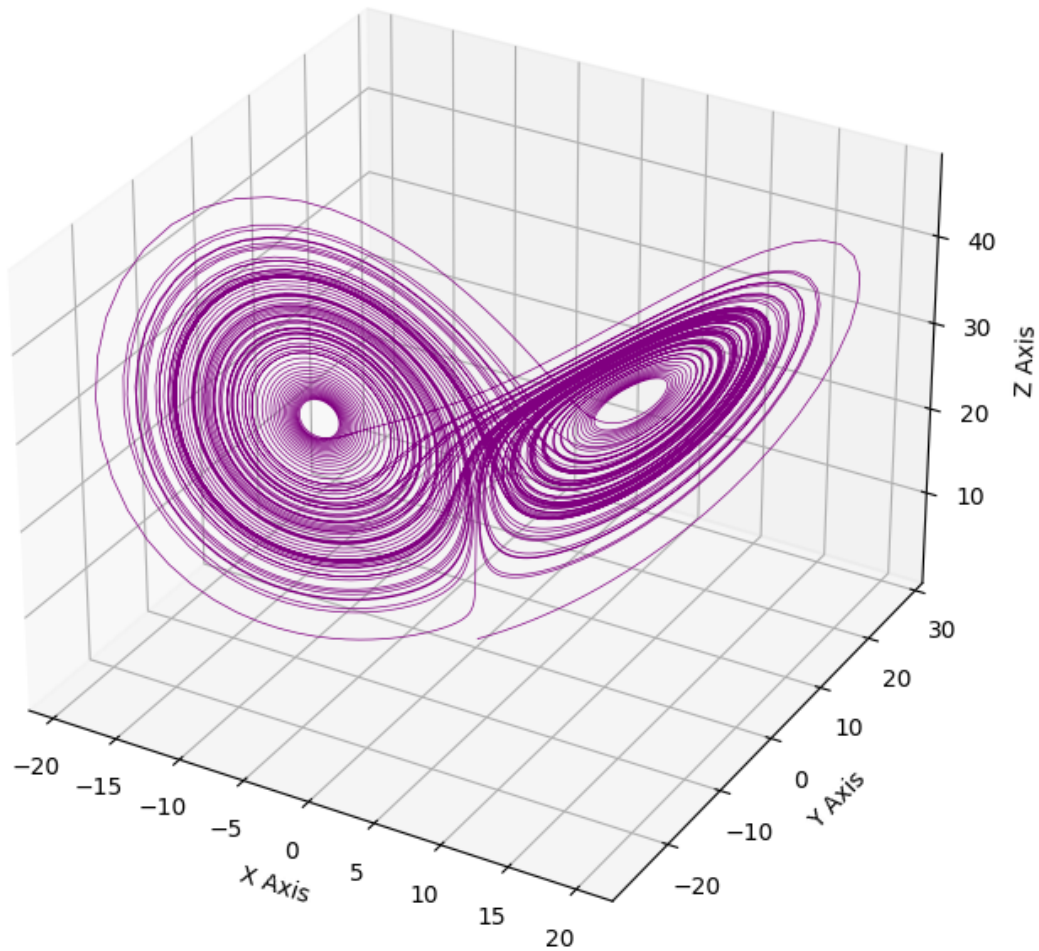


Figure 2.6 Attracteur de Lorenz

3. **Sensibilité aux conditions initiales** : Il existe un $\delta > 0$ tel que, pour tout $x \in A$ et pour tout voisinage V de x , il existe $y \in V$ et $t > 0$ tel que

$$\|\phi_t(x) - \phi_t(y)\| > \delta.$$

4. **Structure géométrique complexe** : L'attracteur a souvent une dimension fractale, ce qui signifie que son apparence et ses détails ne changent pas, quelle que soit l'échelle à laquelle il est observé.

Ces attracteurs sont associés à des comportements complexes et imprévisibles sur le long terme. Un exemple

simple est celui d'un pendule double, pour lequel des trajectoires dans l'espace des états ayant des points initiaux presque identiques vont rapidement diverger. Si l'on tend vers un système plus complexe, la dynamique météorologique en est un autre, avec ses prédictions à long terme difficiles en raison de la sensibilité aux conditions initiales. Ils sont souvent cités comme un exemple d'attracteur étrange, démontrant des comportements complexes et imprévisibles sur une certaine échelle de temps, qui restent néanmoins confinés à des motifs discernables et prévisibles sur des périodes plus courtes.

L'un des exemples les plus célèbres d'un attracteur étrange est l'attracteur de Lorenz (Lorenz, 1963), illustré par la figure 2.6 qui est décrit par un système tridimensionnel gouverné par les équations:

Définition 2.9 (Attracteur de Lorenz)

$$\begin{aligned}\dot{x} &= \sigma(y - x), \\ \dot{y} &= x(\rho - z) - y, \\ \dot{z} &= xy - \beta z.\end{aligned}$$

Où σ , ρ , et β sont des paramètres positifs.

L'attracteur de Lorenz a été tiré d'observations dans les systèmes de convection thermique et est souvent utilisé comme exemple classique de comportement chaotique qui résulte de dynamiques déterministes non linéaires.

2.5.2 La limite du chaos

La limite du chaos est un état spécifique d'un système dynamique où le système opère à la frontière entre l'ordre et le chaos complet. Dans cet état, le système présente une riche variété de comportements dynamiques, capable de répondre de manière flexible et adaptative à des stimuli externes, tout en conservant une structure sous-jacente déterministe. Ce type de dynamiques est caractérisé par l'oscillation aléatoire d'un système entre différents attracteurs limites ou étranges, avec des périodes où le système peut traverser des états chaotiques lors de transition d'un attracteurs semi-stable à un autre.

Des exemples de la limite du chaos peuvent être observés dans divers systèmes naturels et artificiels, tels que les modèles de calcul neuronal, les systèmes écologiques contenant des points de basculement, et

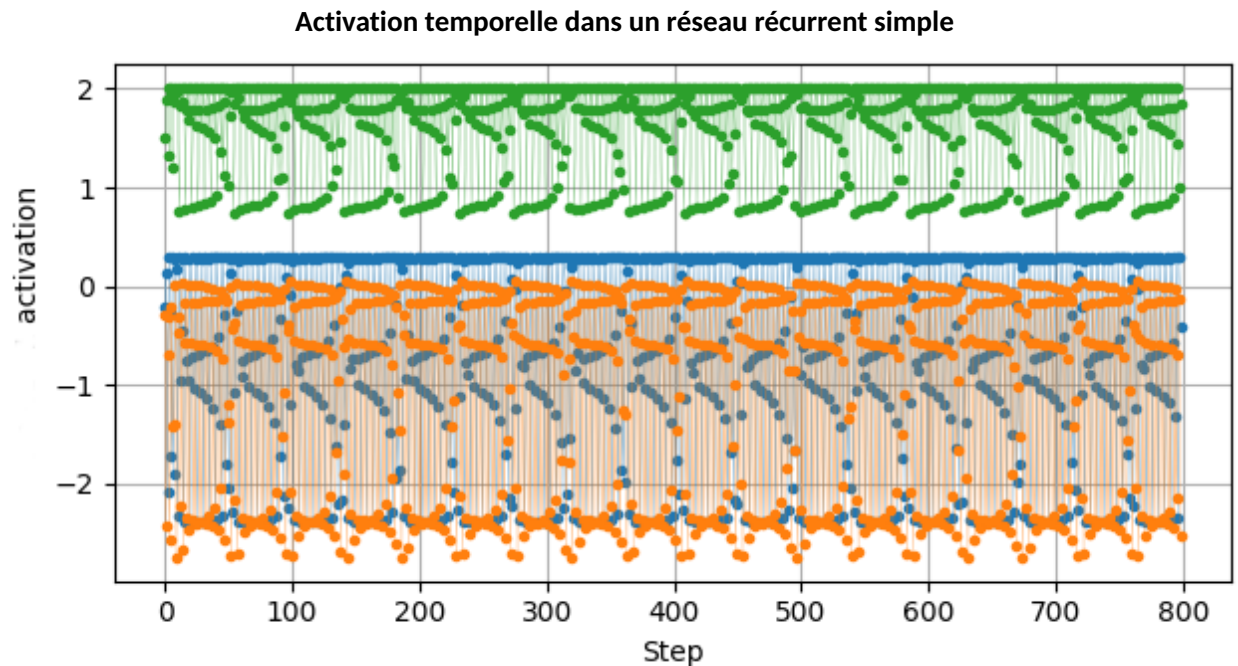


Figure 2.7 Valeurs des états de différents neurones évoluant à travers le temps dans un réseau de neurones récurrents simple. On voit ici qu'il existe un attracteur limite dans un système en raison du motif périodique visible dans ce graphique. L'axe vertical est la valeur du neurone à un temps donné et l'axe horizontal est le nombre d'itérations (ou "step" en anglais) écoulées

dans certains régimes de dynamique des fluides, où de petits changements dans les paramètres peuvent conduire à des transitions soudaines entre des états stables et chaotiques, tel qu'un flot linéaire qui devient turbulent ou lors d'une transition de phase d'un liquide à un gaz.

Au niveau biologique, lors de la résolution de problèmes complexes ou de la créativité, l'activation du cerveau se retrouve souvent à la limite du chaos. Il est théorisé que cet état est exploité pour générer des idées neuves et des solutions innovantes, suggérant que cet équilibre dynamique favorise la flexibilité cognitive et le traitement de l'information (Bilder et Knudsen, 2014).

2.6 Synchronisation

La synchronisation est le processus par lequel des systèmes dynamiques, qu'ils soient naturels ou artificiels, ajustent leurs rythmes ou phases en suite à leurs interactions répétées. Ce phénomène est omniprésent dans la nature, se manifestant dans des systèmes variés tels que des lucioles clignotant à l'unisson, des

Synchronisation d'un système couplé

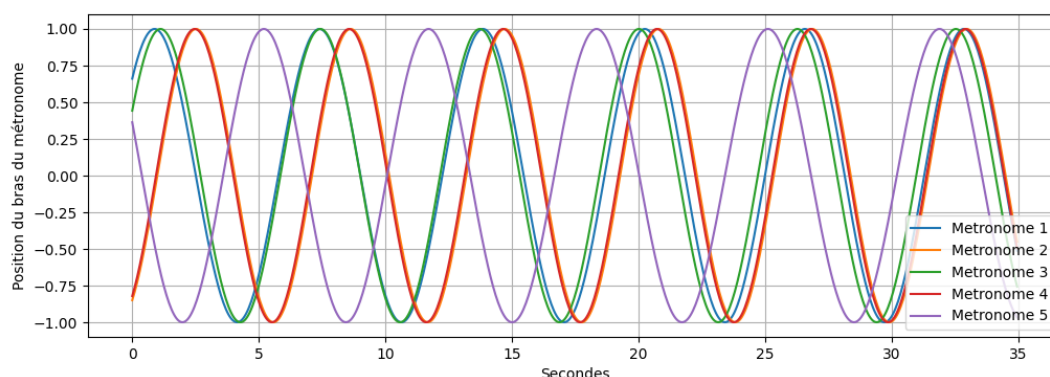


Figure 2.8 Visualisation de l'espace des phases de métronomes (ou oscillateurs) légèrement couplés dans un système simulé

cellules cardiaques battant à l'unisson, des systèmes informatiques distribués et même dans les comportements sociaux humains. Ce comportement dynamique présente souvent des transitions de phase, où un système passe d'un état désordonné (sans synchronisation) à un état ordonné (synchronisation complète) lorsque l'intensité du couplage critique est dépassée. Ce concept est analogue aux transitions de phase en physique, comme le passage de l'état liquide à l'état gazeux.

2.6.1 Métronome

Une expérience simple peut illustrer ce phénomène. Il suffit de prendre plusieurs métronomes mécaniques et de les poser sur une surface instable dans un axe de direction arbitraire y et de poser les métronomes avec leur bras se balançant le long de cet axe. En les démarrant séquentiellement à intervalles arbitraires, on obtient notre système. Peu à peu les différents métronome vont se synchroniser l'un après l'autre en raison de la force exercée sur la surface à chaque changement de direction du bras de chacun d'entre eux (Pantaleone, 2002). Une visualisation graphique de l'espace des phases d'un tel système évoluant dans le temps se retrouve à la figure 2.8

2.6.2 Modèle de Kuramoto

Le modèle Kuramoto est une modélisation mathématique utilisé pour décrire ce type de système. Un ensemble d'oscillateurs couplés et non linéaires, chacun ayant une fréquence propre mais pouvant interagir avec les autres oscillateurs du système. Ce modèle a été proposé par Yoshiki Kuramoto pour étudier la

synchronisation spontanée dans des systèmes complexes (Kuramoto, 1975; Ha *et al.*, 2016).

La dynamique de chaque oscillateur est décrite par l'équation suivante :

$$\frac{d\theta_i}{dt} = \omega_i + \frac{K}{N} \sum_{j=1}^N \sin(\theta_j - \theta_i)$$

Où : - θ_i est la phase de l'oscillateur i , - ω_i est la fréquence naturelle de l'oscillateur i , - K est la force de couplage entre les oscillateurs, - N est le nombre total d'oscillateurs dans le système.

Ce modèle montre que, sous certaines conditions, des oscillateurs d'un système peuvent se synchroniser entre eux. Cela signifie qu'ils atteignent un état où leurs phases se verrouillent et évoluent à la même fréquence collective, malgré une diversité initiale dans leurs fréquences naturelles. Il est largement utilisé pour étudier des phénomènes de synchronisation et trouve des applications dans divers domaines tels que la physique, la biologie, et les neurosciences.

2.6.3 Un cerveau synchronisé

Dans les réseaux récurrents, la synchronisation de groupes de neurones entre eux est essentielle pour l'émergence d'attracteurs dynamiques semi-chaotiques complexes (Berner *et al.*, 2023). Les articles de (Karbowski et Ermentrout, 2002) et (Fischer *et al.*, 2006), pour ne citer qu'eux, se sont intéressés à la synchronisation à l'intérieur de différents oscillateurs neuronaux. Il ressort de ces recherches que la synchronicité est atteignable dans un premier temps avec des réseaux récurrents aléatoires puis dans un second dans des réseaux avec de l'apprentissage Hebbien pour autant que les poids inhibiteurs/excitants demeurent balancés. Un élément étonnant qui émerge également des différentes études est la rapidité avec laquelle les oscillateurs se synchronisent; pour les oscillateurs dont les forces synaptiques sont fixées, la synchronisation prend dans plusieurs cas une fraction du nombre d'itérations contenu dans un cycle complet; lorsque l'apprentissage Hebbien entre en jeu, il s'agit généralement de quelques cycles seulement pour atteindre une certaine stabilité des dynamiques et des poids synaptiques.

CHAPITRE 3

LE CERVEAU

L'intérêt pour la recherche en intelligence artificielle remonte jusqu'aux balbutiements des sciences informatiques, qui ont vu le jour en 1936 avec la première machine de Turing. Dès 1943, McCulloch et Pitts détaillaient la première formulation du perceptron (McCulloch et Pitts, 1943), l'unité fonctionnelle de base dans les réseaux de neurones modernes. Le père fondateur de l'informatique, Turing lui-même, parlait de "machine intelligente" en se référant à son invention et s'attela dès 1948 à concevoir des algorithmes d'apprentissage machine basés sur le fonctionnement du cerveau (Turing, 2004). L'homme derrière l'architecture de l'ordinateur contemporain, Von Neuman, dans son livre séminal "L'ordinateur et le cerveau" (Neumann *et al.*, 2000)¹ parut en 1958, s'attarde à dresser une étude comparative entre l'ordinateur et le cerveau et à suggérer des pistes de recherche bio-inspirées pour l'avancement de l'IA.

Depuis cette époque, les chercheurs ont perpétué cette tendance de s'inspirer de mécanismes biologiques (Hopfield, 1982; Lobo *et al.*, 2020), fonctionnels (Feldman, 2012; Macpherson *et al.*, 2021), topologique (Macpherson *et al.*, 2021), neurologiques (Krotov et Hopfield, 2019; Macpherson *et al.*, 2021) et psychologiques (Anderson, 2007) pour développer de nouveaux algorithmes "intelligents".

Depuis le perceptron avec la propagation arrière, de nombreuses autres techniques développées au fil des ans ont continué à être largement inspirées par l'apprentissage neurologique, avec la mention explicite d'inspiration biologique dans de nombreux articles proposant des variations ou de nouveaux algorithmes d'apprentissage machine.

Cette riche histoire commune entre les sciences cognitives et l'IA, l'intérêt des plus grands penseurs des sciences informatiques pour le cerveau et le rythme constant des découvertes en neuroscience propulsé par les avancées continues en imagerie cérébrale, en biologie et algorithmiques nous pousse à croire que l'intersection entre ces deux domaines est la voie à suivre pour espérer atteindre un jour l'intelligence artificielle générale (AGI, de l'anglais *artificial general intelligence*).

Les transformeurs (Vaswani *et al.*, 2017), qui sont présentés à la section 4.1.2.3 représentent l'état de l'art en termes de capacités d'IA largement applicables. Cette technologie, à laquelle on doit les avancées fulgurantes sur le traitement du langage naturel des dernières années, s'est révélée après coup avoir un fonctionnement analogue à une zone de l'hippocampe, décrit ci-bas en 3.1 (Whittington *et al.*, 2022). Cette découverte renforce une fois de plus la dépendance de l'IA sur la neuroscience.

¹ Traduction libre, de l'anglais "The computer and the brain" (Neumann *et al.*, 2000)

Ce chapitre vise à présenter les différentes notions neurologiques tirées de l'étude de celui-ci, humain et animal, qui nous semblent pertinentes pour l'élaboration d'une modélisation d'un réseau dynamique fonctionnel présentant des capacités d'optimisation tout en ayant des caractéristiques dynamiques analogues à l'apprentissage biologique.

3.1 L'hippocampe

L'hippocampe est une structure critique du cerveau humain que l'on retrouve dans le lobe temporel du cerveau. Il est largement considéré comme une pièce centrale du système de création et de rappel de mémoires sémantiques et épisodiques. Par le fait même, il est une zone centrale du processus d'apprentissage à partir de nouvelles expériences. Il est également considéré comme étant une structure centrale pour la consolidation des mémoires à court terme en mémoire à long terme (Witter, 2012).

Son fonctionnement est intimement lié à celui d'autres structures du système limbique, contribuant à la régulation des émotions et des comportements motivés (Anand et Dhikav, 2012). L'une de ses fonctions principales de l'hippocampe est l'encodage contextuel des stimuli sensoriels interprétés et des lieux, permettant aux individus de se repérer dans leur environnement et de se remémorer des événements spécifiques (mémoire épisodique) (Witter, 2012). Cela démontre le rôle central de la mémoire pour l'émergence de comportements intelligents.

3.1.1 Organisation des neurones dans l'hippocampe

L'hippocampe comporte trois régions principales, CA1, CA2 et CA3. L'étude de la troisième, CA3, est particulièrement intéressante dans notre contexte puisque, comportant de nombreuses connexions récurrentes vers elle-même et vers les deux régions inférieures, via un réseau dense de cellules pyramidales interconnectées (Witter, 2012) son graphe de connectivité en fait une forme de RNN. Ces connexions forment un circuit auto-associatif, permettant à la région CA3 de jouer un rôle crucial dans le stockage et le rappel d'information (Bennett *et al.*, 1994). Son organisation unique facilite la formation de motifs d'activité spécifiques, ou "ensembles de neurones" s'activant de façon ordonné et périodique, qui représentent l'encodage des éléments ou des aspects d'expériences mémorisées. Ce circuit a la capacité de sauvegarder ces motifs à l'intérieur d'attracteurs limites pour un rappel ultérieur même après une seule exposition à un stimuli distinct.

Dans le domaine de l'intelligence artificielle plus spécifiquement, la structure et la fonction de la région

CA3 de l'hippocampe ont inspiré le développement d'algorithmes pour l'apprentissage de mémoires dans des réseaux de neurones récurrents dès 1982 (Hopfield, 1982). De nombreuses recherches subséquentes ont tenté de reproduire les mécanismes de l'apprentissage auto-associatif et la capacité de généralisation observés dans CA3 (Hopfield, 1984; Gosti *et al.*, 2019; Lobo *et al.*, 2020; Macpherson *et al.*, 2021), qui nous offrent des pistes pour la création de systèmes capables d'apprendre de manière autonome à partir de séquences d'événements et de se remémorer des informations de manière contextuelle. Pour ne citer qu'un exemple concret, les modèles de mémoire associative basés sur les principes de fonctionnement de CA3 sont explorés afin d'améliorer la reconnaissance de motifs, la prédiction de séquences et la navigation autonome dans des environnements complexes (Lobo *et al.*, 2020; Macpherson *et al.*, 2021).

3.2 Le cerveau comme système dynamique

Le cerveau humain peut être représenté comme un système dynamique de la forme $S_{t+1} = f(S_t, I_{\text{sensorielle}}, \gamma)$, où l'état des neurones S au temps t est fonction de leur état précédent, des entrées sensorielles et γ un terme général qui englobe toutes les spécificités biochimiques ayant un impact sur l'état des neurones. Cette formalisation est désormais fréquemment utilisée par les chercheurs pour créer des modèles de RNP et de RNN afin d'étudier l'impact de la variation des différents paramètres appartenant abstrait dans γ , tel que différents neurotransmetteurs (Holca-Lamarre *et al.*, 2017; Pickering et Pesola, 2014; Graupner et Gutkin, 2009; Yin et Wang, 2016).

3.2.1 Attracteur pour la navigation spatiale

Dans la région CA1 de l'hippocampe, il existe des neurones uniques qui génèrent des influx nerveux lorsqu'un sujet se trouve à un endroit précis dans l'espace appelé "neurone de lieu" (en anglais "place-cells"). Un autre type de neurones, toujours dans cette région s'activera lorsque le sujet pose des actions qui auront pour effet de modifier son emplacement dans l'espace, ou même l'espace en lui-même (O'Keefe, 1976). Concrètement, cela suggère qu'un neurone précis du premier type s'active chaque fois que nous nous trouvons sur le seuil d'une porte et un neurone du second type s'active si nous ouvrons cette porte.

Plus récemment, les recherches de (Hafting *et al.*, 2005) ont révélé que dans le cortex entorhinal dorsomédial, qui est fortement connecté à CA1, il existe des neurones qui s'activent périodiquement lorsque le sujet se trouve à des endroits précis dans l'espace. Ces neurones sont appelés "neurones de grilles", car si l'on cartographie les emplacements où l'une de ces cellules est activée alors qu'un rat se déplace dans un espace

fini (par exemple, une boîte), ceux-ci s'activent à des intervalles réguliers pour former un motif de grille triangulaire.

Ce type d'activation périodique contextuelle démontre donc la présence d'une oscillation sur un point fixe, si l'on tient compte seulement du neurone en question, et d'attracteurs cycliques, si l'on prend en compte la suite complète d'activations menant au potentiel d'action du neurone observé, dans le cerveau. Nous pouvons également en retirer que ces attracteurs font partie des éléments de base constituant le modèle interne qui est créé par le cerveau pour traiter l'information et générer des actions.

3.2.2 Bruit stochastique

Comme tout ingénieur en fait l'expérience rapidement dans sa carrière, le monde physique est désordonné et il existe un écart fondamental dès lors que l'on tente d'appliquer la théorie expérimentalement. Le cerveau n'échappe pas à cette règle, et bien que les neurones suivent des règles bien précises qui dictent leur dynamique, il arrive qu'ils ne s'activent sans que celui-ci ne reçoive suffisamment de potentiels d'actions pour le justifier. Également, il arrive que malgré des stimuli entrants suffisants, un neurone fasse défaut et ne produise pas d'influx nerveux alors que toutes les conditions sont réunies pour que ce soit le cas (Faisal *et al.*, 2008).

Le cerveau contient ainsi par défaut un certain bruit stochastique de fond. Toutefois, il est maintenant reconnu que ce bruit peut avoir des effets positifs pour les calculs effectués dans le cerveau, tel que l'augmentation de signaux faibles via un processus appelé la résonance stochastique (Faisal *et al.*, 2008). Également, il a été prouvé que la présence de bruit au moment de l'entraînement de réseaux de neurones profond permet l'apprentissage d'une représentation plus précise et robuste par ce dernier (Neelakantan *et al.*, 2015), ainsi qu'une réduction du surapprentissage (Noh *et al.*, 2017). Ce bruit stochastique est potentiellement ainsi non pas une limitation du système mais bien un avantage au niveau computationnel. Ceci expliquerait également qu'il soit toujours présent après 521 millions d'années d'évolution (Strausfeld *et al.*, 2016), suggérant également qu'il s'agisse d'un mécanisme finement optimisé.

3.2.3 Le cerveau à la limite du chaos

Selon le modèle développé dans les recherches de (Kitzbichler *et al.*, 2009), le cerveau opère à la limite du chaos, ses dynamiques internes alternant entre des périodes prolongées en verrouillage de phase et

de changements rapides, exploitant cet état pour optimiser son traitement de l'information, sa flexibilité cognitive et sa capacité à transiter rapidement entre différents états fonctionnels. Les travaux de Knudsen suggèrent également que la limite du chaos est essentielle pour la créativité (Bilder et Knudsen, 2014). Ils proposent que des dynamiques malsaines puissent expliquer certains des dysfonctionnements associés à des conditions psychologiques telles que la schizophrénie, dans le cas où les dynamiques seraient trop chaotiques, ou le syndrome d'Asperger, dans le cas où elles seraient trop ordonnées. On peut en déduire que le changement entre les différents attracteurs dominant constitue un point critique essentiel pour une cognition saine et productive et qu'il existe un délicat point d'équilibre optimal vers lequel le cerveau doit tendre pour effectuer un traitement efficace de l'information. Suivant les modélisations de (Steyn-Ross *et al.*, 2014), le cerveau exhiberait pareillement ce régime dynamique lors des phases de sommeil profond, dénotant son importance pour diverses fonctions cérébrales.

3.3 Apprentissage en ligne en un coup

Nous pouvons tous ramener dans notre esprit des souvenirs d'événements marquants issus d'une époque révolue. Que ce soit une graduation, un mariage ou l'odeur de notre première voiture neuve à son premier jour, les données sensorielles associées à ces événements n'ont été présentées à notre cerveau qu'une seule fois et, pourtant, leur seule évocation est suffisante pour les ramener en mémoire. Signifiant qu'il est possible d'y accéder même une fois que plusieurs années se sont écoulées, quelques fois des décennies, même suite aux nombreux changements plastiques qui sont survenus dans le cerveau par le biais de nouveaux souvenirs et de consolidation d'apprentissages. Ce type de mémoire pérenne illustre bien l'une des capacités exceptionnelles de notre cerveau, l'apprentissage en un coup pérenne d'un stimulus.

Un autre exemple de cette capacité de mémorisation à partir d'une seule exposition est notre capacité à reconnaître une personne que nous n'avons vue qu'une seule fois ou de retrouver notre chemin dans un environnement où nous ne nous sommes rendus qu'une seule fois. Mais notre capacité d'apprentissage en un coup ne se limite pas à la mémorisation. Les travaux de (Tiedemann *et al.*, 2022) dévoilent l'existence d'un modèle génératif interne qui permet de créer des variations appartenant à une catégorie donnée à partir d'une seule image de cette catégorie.

Différentes études ont également révélé qu'il est possible pour des sujets de se remémorer des séquences d'images qui ne leur avaient été présentées qu'une seule fois sur des périodes allant de quelques heures à quelques jours (Jacques *et al.*, 2013; MacLeod *et al.*, 2018).

La question de comprendre les mécanismes impliqués qui rendent possible une telle faculté en est une qui préoccupe les neuroscientifiques et est le sujet de nombreuses recherches (Turner et Onysk, 2022; Lee *et al.*, 2015). Elle comporte deux volets: l'apprentissage en un coup et la rétention sur le long terme. Ces deux catégories ont fait l'objet de nombreuses études et cette section vise à en présenter certaines qui sont pertinentes pour le présent travail de recherche.

3.3.1 Apprentissage en ligne à long terme

Shaw et Porter ont démontré que des souvenirs pouvaient être modifiés via le rappel (Shaw et Porter, 2015). Dans leur expérience, les sujets étaient appelés à se remémorer des souvenirs semi-lointains et les chercheurs mentionnaient un élément marquant fictif, tel qu'une arrestation en énonçant le souvenir. Tous les sujets ont à ce moment relevé que cet élément n'avait pas pris place dans le souvenir en question. Cependant, lorsque quelques mois plus tard lors d'une seconde rencontre, les sujets étaient appelés à relater le souvenir, celui-ci incluait l'élément fictif pour une partie des participants. Cela est induit par le fait que la mémoire ne fonctionne pas comme un enregistreur, mais s'applique plutôt à reconstruire les événements à partir d'un sous-ensemble compressé de l'information original (Loftus, 1996), soutenant également l'existence d'un modèle génératif interne permettant de décompresser une mémoire encodée. Ces faux souvenirs causent de nombreux problèmes, notamment au niveau du système de justice: "Le nombre d'erreurs d'identifications conduisant à des accusations erronées, combiné au fait que le témoignage oculaire est accepté trop facilement par des jurys, est un réel problème pour la justice"²

Ces expériences soulignent la nature "en ligne" du cerveau, où la limite entre l'apprentissage et le rappel est floue, tel que corroboré par de nombreux travaux subséquents (Jacques *et al.*, 2013; Siestrup et Schubotz, 2023), et qu'elle est même inexistante dans certains cas (Censor *et al.*, 2010).

Dans le cas d'expériences traumatiques, comment la thérapie peut aider à recadrer des souvenirs afin qu'ils soient mieux intégrés dans le but qu'ils cessent de causer du tort au patient (Kar, 2011) est un exemple positif des outils thérapeutiques qui utilisent ce principe neurologique. Cependant, sans ce travail cognitif d'accès et d'édition, le souvenir peut demeurer inchangé sur de très longues périodes. Cela implique un mécanisme neurologique qui régule à un haut niveau la plasticité de certains neurones en fonction des informations que leurs synapses encodent et de leur positionnement topologique dans le graphe de calcul

² Également tiré de (Loftus, 1996), p. 201, traduit de l'anglais par Pierre de Sutter

cérébral (Larzabal *et al.*, 2018).

3.4 Activation binaire

La signalisation neuronale et le métabolisme énergétique sont intrinsèquement liés aux propriétés biochimiques des cellules nerveuses. La neurotransmission, reposant sur la libération et la réception de neurotransmetteurs dans la fente synaptique, est régulée par une série de réactions chimiques spécifiques. Ces processus sont étroitement liés à la disponibilité de substrats énergétiques comme le glucose et l'oxygène, ainsi qu'à la disponibilité des neurotransmetteurs dans les terminaisons synaptiques, éléments nécessaires pour une activité neuronale soutenue. Ces ressources étant limitées, les travaux de (Macpherson *et al.*, 2021) suggèrent qu'une activation binaire permettrait une gestion plus efficace des ressources énergétiques.

Le voltage de la membrane des neurones comporte donc deux états principaux en ce qui attrait à la transmission de l'information. Lorsqu'elle se trouve sous le potentiel seuil, généralement entre -50mV et -55mV, elle est désactivée et très peu d'informations sont transmises aux neurones en aval (Hammond, 2015). Lorsque le seuil est atteint, les canaux ioniques Na⁺ s'ouvrent, dépolarisant la membrane en laissant entrer les ions Na⁺ extracellulaires jusqu'à atteindre un voltage d'environ 30mV. Le neurone à ce moment est activé et l'influx nerveux est propagé par l'axone aux synapses qui libèrent ensuite leurs vésicules de neurotransmetteurs aux neurones subséquents dans le graphe computationnel. Par la suite, les canaux ioniques Na⁺ se referment alors que ceux K⁺ s'ouvrent, créant un flot d'ions de potassium qui quittent la cellule, re-polarisant par le fait même la membrane jusqu'à son état initial et le dépassant dans le cas où le neurone est peu myéliné (Hammond, 2015).

3.4.1 Réfraction

Suite à son activation, un neurone tombe en état de réfraction pour une période de quelque millisecondes. Pendant cette période, il ne peut se réactiver à nouveau et est considéré comme insensible aux stimuli entrants. Cela évite dans un premier qu'un neurone soit trop activé à répétition et dans un second, cela laisse le temps aux neurones en aval d'intégrer son signal sans qu'il n'y ai possibilité de confusion entre deux pics subséquents. Cette période permet également au neurone de revenir à son équilibre biochimique, au niveau de la concentration d'ions à l'interne, évitant par le fait même des impulsions ambiguës (dans le cas où le voltage du neurone lors du pic ne pourrait pas atteindre sa valeur habituelle).

3.4.2 Encodage de l'intensité dans la fréquence

Cependant, ce n'est pas parce que les impulsions des neurones sont binaires que celles-ci ne sont pas en mesure d'encoder une gradation du stimuli d'entrée. Bien que les neurones génèrent des potentiels d'action de manière tout ou rien, à amplitude fixe, l'intensité d'un stimulus est encodée via la fréquence de ces potentiels d'action. Ce phénomène, connu sous le nom d'encodage fréquentiel, permet au système nerveux de transmettre des informations sur l'amplitude du stimulus (Tabas et von Kriegstein, 2021). La fréquence des pics d'activations augmente avec l'intensité du stimulus, permettant ainsi une modulation fine de l'activité neuronale en réponse à l'environnement dans un contexte où l'amplitude de l'activation en elle-même n'est pas un paramètre.

3.5 Règle d'apprentissage locale

L'intelligence peut être vue comme un phénomène émergent résultant de l'interaction complexe entre de nombreux éléments simples. Dans des systèmes tels que les réseaux de neurones, l'intelligence n'est pas attribuable à un élément unique mais émerge de l'organisation collective et de la dynamique des interactions neuronales. Cette perspective suggère que l'intelligence est en partie le produit de règles locales et simples opérant à l'échelle des composants individuels, engendrant des comportements complexes et adaptatifs à une échelle supérieure.

L'intelligence des fourmis est un exemple classique d'intelligence collective et émergente qui est basée sur des règles simples et locales. Sans une supervision centrale, les fourmis parviennent à résoudre des problèmes complexes, comme la recherche de nourriture ou la construction et la défense de la fourmilière, à travers des interactions locales entre individus (Millonas, 1992; DAVID MORGAN, 2009). Ces comportements sont régis par des règles simples d'interaction et de communication basées, entre autres, sur le dépôt de phéromones (Czaczkes et Heinze, 2015; DAVID MORGAN, 2009), qui guident collectivement le groupe vers des solutions efficaces (Gelblum *et al.*, 2020).

3.5.1 Plasticité synaptique temporelle

La plasticité synaptique dépendante du temps de l'activité neuronale (STDP, de l'anglais spike-time dependent plasticity), est un mécanisme fondamental d'apprentissage et de mémorisation dans les réseaux de neurones biologiques. Elle réfère à la modification de la force des connexions synaptiques en fonction du temps

d'occurrence relatif des potentiels d'action entre les neurones pré- et post- synaptiques (Buonomano et Carvalho, 2009).

La règle de base de la STDP stipule que si un neurone pré-synaptique s'active avant un neurone post-synaptique (pré-puis-post), cela renforce la synapse, facilitant la transmission future des signaux. Inversement, si le neurone post-synaptique a une pointe avant le neurone pré-synaptique (post-avant-pré), la force synaptique est atténuée, rendant la transmission ultérieure de signaux plus difficile (Feldman, 2012).

La STDP est également influencée par la synchronicité des rafales de potentiels d'activation. Des rafales synchronisées entre neurones pré- et post-synaptiques peuvent renforcer significativement les synapses, favorisant l'émergence de groupes neuronaux synchronisés (Anisimova *et al.*, 2022) essentiels pour la formation de réseaux fonctionnels et pour l'encodage de l'information (Palva *et al.*, 2010; Rolls et Treves, 2011)

3.5.1.1 Facteurs additionnels

Comme un grand nombre de processus biologique acquit à travers l'évolution, il existe une multitude de facteurs additionnels qui ont un impact sur la potentiation des neurones et la plasticité. Les concentrations intra- et extracellulaires des différents ions positifs et négatifs responsables de la variation du voltage de la membrane (Barreto et Cressman, 2011), la disponibilité dans la terminaison synaptique des vésicules contenant les neurotransmetteurs requis pour la transmission des impulsions en aval (Lou *et al.*, 2012) et même la quantité de sous-produits indésirables des réactions chimiques impliqués dans la transmission d'influx nerveux telle l'amyloïde (Karisetty *et al.*, 2020) sont tous des éléments qui viennent moduler les caractéristiques fonctionnelles des neurones en temps réel. Ces éléments sont des champs de recherche actifs extrêmement complexes en raison du grand nombre d'interactions impliquées et dépassent largement le cadre des travaux présentés ici. Nous nous en tenons par conséquent à la forme simplifiée de la STDP.

3.5.1.2 Algorithme

Formellement l'algorithme simplifié de la STDP biologique peut être exprimé comme suit (algorithme tiré de (Song *et al.*, 2000)):

Soit w_{ij} la capacité de transmission des synapses du neurone i aux dendrites du neurone j . Le changement synaptique Δw_{ij} suite à une activation peut être modélisé :

$$\Delta W = \begin{cases} A_+ \times \exp\left(\frac{t_{pre}-t_{post}}{\tau}\right), & \text{si } t_{post} > t_{pre} \\ A_- \times \exp\left(\frac{t_{post}-t_{pre}}{\tau}\right), & \text{si } t_{post} < t_{pre} \end{cases} \quad (3.1)$$

où W représente les poids synaptiques, A_+ et A_- sont les taux d'apprentissage, t_{pre} et t_{post} sont les temps d'occurrences des impulsions des neurones pré et post-synaptiques, respectivement, et τ est la constante de temps.

3.6 Autres facteurs

3.6.1 Alpha, Beta, Theta, Delta

Il existe différentes fréquences caractérisant les propriétés fonctionnelles et dynamiques du cerveau. Celles-ci sont généralement reconnues comme jouant un rôle fondamental dans la modulation et la coordination de l'activité neuronale. Ces rythmes sont cruciaux pour une variété de fonctions cognitives et reflètent différents états de conscience. Leur modulation pendant les processus d'apprentissage et de mémorisation suggère que les différents régimes cérébraux contribuent à l'optimisation et à la consolidation du traitement de l'information.

Les ondes bêta (14-30Hz) sont les plus rapides et reflètent des états d'alerte, de concentration et de cognition active. Elles sont caractéristiques des périodes de prise de décision, de résolution de problèmes et d'autres activités mentales intensives (nuryadi *et al.*, 2020). Une augmentation de l'activité bêta vers son régime plus élevé est souvent liée à l'anxiété ou à l'excitation (Ribas *et al.*, 2018).

Les ondes alpha (8-13Hz) sont particulièrement présentes dans un état de relaxation avec les yeux fermés, mais elles diminuent lors de l'ouverture des yeux ou de la concentration mentale (Moini et Piran, 2020). Elles sont souvent associées à des états de calme, de méditation (katyal et goldin, 2021), et servent de pont entre la conscience et le subconscient, facilitant la créativité (Fink et Benedek, 2014) et la mémorisation (Makada *et al.*, 2016).

En continuant vers les fréquences plus basses, nous arrivons aux ondes thêta (4-8 Hz) qui sont typiquement associées aux états de sommeil léger, ou de relaxation profonde (Moini et Piran, 2020). Les ondes thêta jouent un rôle crucial dans les mécanismes de plasticité cérébrale (tsanov et manahan vaughan, 2009;

tang *et al.*, 2019), l'intégration sensorielle et sont par conséquent importantes pour l'apprentissage, la mémorisation et l'intuition.

Finalement, les ondes delta (0.1-4Hz) sont quant à elles principalement observées pendant le sommeil profond sans rêves et chez les très jeunes enfants durant le développement du cerveau (Moini et Piran, 2020). Ce sont les plus lentes, présentent la plus grande amplitude et sont synonymes de régénération, de réparation autant au niveau corporel (hoda elkhenany *et al.*, 2018) que cérébral (nina e. fultz *et al.*, 2019).

On observe ici qu'un ralentissement des ondes cérébral est associé avec des états progressivement plus détendus et également un apprentissage plus prononcé et profond, alors qu'une accélération tend à engendrer des états d'alerte progressivement plus accentués.

3.6.2 Neurotransmetteurs

Les neurotransmetteurs, comme le glutamate, le GABA, la dopamine, et la sérotonine, modulent l'activité neuronale et influencent des processus cérébraux variés tels que l'apprentissage (Tellez *et al.*, 2012b), la motivation et l'adaptabilité comportementale (Peters *et al.*, 2021). Ils agissent comme des messagers chimiques, facilitant ou inhibant la transmission synaptique (Reis *et al.*, 2009), et leur concentration et libération sont ajustées en réponse à l'activité neuronale et aux exigences de l'apprentissage dans différents contextes (Tellez *et al.*, 2012a; Uddén *et al.*, 2010; Olvera-Cortés *et al.*, 2008; Wert-Carvajal *et al.*, 2022), soulignant leur rôle primordial dans la modulation de la plasticité synaptique. Ceux-ci permettent une modulation en temps réel à la fois globale (Decot *et al.*, 2017) et sélectives (Gao et Goldman-Rakic, 2003) des différents circuits neuronaux à travers le cerveau.

Dans le contexte de notre recherche, nous retenons ici simplement que malgré les règles d'activation locales mentionnées plus haut, la bio-plausabilité n'exclut pas nécessairement une forme de communication diffuse plus globale ayant un effet global sur le réseau.

3.7 Organisation des neurones dans le cortex en colonnes

Dès 1957, les recherches de Mountcastle ont permis de découvrir l'unité structurelle et fonctionnelle fondamentale du cortex cérébral, la minicolonne. Ces colonnes ont une composition et une organisation similaires les unes par rapport autres, constituées d'environ 80 à 110 neurones disposés de façon cylindrique avec une

orientation perpendiculaire à sa surface qui est répétée à travers le cerveau, de façon à former un pavage sur le plan couvrant le cortex (Mountcastle, 1957). Densément interconnectés, les neurones la composant ont un champ réceptif similaire et travaillent de concert pour traiter un influx nerveux distinct. C'est pourquoi elles sont considérées comme les briques élémentaires du traitement de l'information dans le cerveau, chaque colonne se spécialisant dans le traitement d'un type spécifique d'information sensorielle ou cognitive (Molnár, 2013).

Les mini-colonnes sont organisées en groupe variant de 50 à 100 minicolonnes appelées hypercolonnes. Cette architecture permet de faciliter le filtrage, qui est à la base du principe de l'attention, et l'organisation interne permet un traitement hiérarchique de l'information. Aussi, les travaux de (Hubel et Wiesel, 1968) soutiennent qu'il existerait aussi une organisation hiérarchique entre les hypercolonnes elles-mêmes et il a été avancé que cette hiérarchie de traitements est l'un des éléments clés dans l'organisation et l'abstraction d'informations complexes à travers le cerveau (Peissig et Tarr, 2007) et ce modèle a été reproduit informatiquement pour réaliser la prédiction et la planification complexe impliquant différentes échelles de temps (Niu *et al.*, 2022).

3.7.1 Architecture modulaire

L'arrangement du cerveau en un graphe de traitement topologiquement hiérarchique (Hubel et Wiesel, 1968) est donc facilité par l'organisation du cerveau en unités fonctionnelles distinctes, capables de traiter des informations de manière semi-indépendante. Chaque minicolonne, ou module, étant spécialisée dans un type spécifique de traitement d'information, cela permet une parallélisation et une intégration efficace des tâches cognitives (Peissig et Tarr, 2007). Cette modularité favorise aussi la robustesse du système neural, car les dysfonctionnements localisés peuvent être isolés sans perturber l'ensemble du réseau. C'est souvent le cas d'ailleurs lors de la perte de stimuli sensoriels dû, par exemple, à une amputation ou à un problème neurologique que les zones qui étaient dédiées au traitement de cette information se réorganisent pour traiter de nouveaux stimuli évitant de devenir inutilisées et démontrant la compétition entre les différents stimuli pour leur traitement (Knecht *et al.*, 1998). Inversement, lorsqu'une région du cerveau est endommagée, la topologie du graphe cérébral se modifie de façon à rediriger les informations traitées par la zone détériorée vers une zone saine (Jones et Adkins, 2015).

3.7.2 Dynamiques internes

Les dynamiques internes des colonnes neuronales et des modules cérébraux sont régies par des interactions complexes entre les neurones au sein d'une même couche de la colonne, entre les différentes couches et entre les différentes colonnes. Ces dynamiques sont caractérisées par des boucles de rétroaction et des mécanismes d'inhibition latérale qui permettent d'ajuster et de stabiliser l'activité neuronale en réponse à des stimuli internes et externes (Hubel et Wiesel, 1968) pour converger sur des attracteurs significatifs (Peissig et Tarr, 2007).

3.8 Mémoire associative contextuelle

La mémoire associative contextuelle se réfère à la capacité du cerveau à former et récupérer des souvenirs basés sur les associations entre des stimuli et leur contexte. Elle implique l'intégration d'informations provenant de différentes sources sensorielles et leur stockage dans une représentation cohésive qui peut être activée par des indices contextuels. Ce type de mémoire est essentiel pour naviguer dans des environnements complexes et pour la prise de décision, car elle permet aux individus de se rappeler des expériences passées dans des contextes similaires et d'utiliser ces souvenirs pour guider les comportements futurs.

3.8.1 Intégration globale de l'information

Une recherche qui exprime bien cette intégration globale de l'information est celle effectuée par (Williams et Bargh, 2008). L'expérience commençait dans l'ascenseur, à l'insu du sujet, alors qu'un complice demandait au sujet si ce dernier pouvait tenir son breuvage le temps que le complice prenne des notes. Les sujets étaient divisés en deux groupes: le premier groupe se faisait tendre un breuvage chaud (café) et le second un breuvage froid (café glacé). Plus tard dans l'expérience, les chercheurs demandaient aux sujets de décrire la personne qu'ils avaient croisée dans l'ascenseur. Le groupe ayant tenu le breuvage chaud était plus enclin à décrire la personne dans l'ascenseur plus positivement, donc en utilisant des adjectifs comme chaleureuse, généreuse, sociale ou plus joyeuse, que ceux qui avaient tenu le breuvage froid. On peut voir ici comment un stimulus sensoriel de bas niveau, la température au niveau de la main, qui est traité de façon inconsciente vient néanmoins impacter la perception de la réalité du sujet sans que celui-ci en soit conscient.

CHAPITRE 4

ALGORITHMES

Maintenant que nous avons établi les différents éléments théoriques en neurosciences qui supportent les décisions faites dans la formulation de notre proposition, nous nous tournons maintenant vers les différentes approches algorithmiques proposées dans la littérature sur l'apprentissage machine. Le présent chapitre vise à offrir un survol des techniques les plus pertinentes pour le cadre de notre recherche.

4.1 Propagation arrière

La première technique est également la plus connue et la plus utilisée de celles que nous avons retenues. Il s'agit de la propagation arrière, ou rétropropagation. Il n'y a pas de consensus officiel sur son origine puisque différents concepts s'en approchant et s'y rattachant ont été définis à plusieurs moments sous différentes formes plus ou moins abouties. Nous pouvons néanmoins retracer la première utilisation du terme "correction d'erreur via la propagation arrière" à (Rosenblatt, 1963) en 1962. Huit ans plus tard Seppo Linnainmaa publie la version moderne telle que nous la connaissons aujourd'hui (Linnainmaa, 1970).

La RP applique le théorème de dérivation des fonctions composées de Leibniz. Elle estime le gradient des différents poids dans un réseau de neurones afin de les optimiser itérativement par rapport à l'erreur calculée à partir de la différence entre une prévision en sortie du réseau sur un vecteur d'entrée et la sortie réelle attendue pour cette entrée. Cette différence est calculée à l'aide de la fonction d'erreur, qui varie en fonction du problème que l'on tente de résoudre. Cette fonction est le terme que l'on tente d'optimiser au moment de l'entraînement d'un réseau de neurones.

4.1.1 Algorithme original

L'algorithme est composé de quatre éléments principaux: la propagation avant, le calcul de l'erreur, la rétropropagation de l'erreur et la mise à jour des poids. Formellement:

Définition 4.1 (Rétropropagation) 1. Propagation avant

Pour chaque couche $l = 1, \dots, L$ du réseau, où L est la dernière couche, calculer les activations :

$$a^{[l]} = f^{[l]}(z^{[l]}) \quad \text{où} \quad z^{[l]} = W^{[l]}a^{[l-1]} + b^{[l]}$$

avec $a^{[0]}$ étant l'entrée du réseau, $W^{[l]}$ et $b^{[l]}$ étant respectivement les poids et les biais de la couche l , et $f^{[l]}$ la fonction d'activation pour la couche l .

2. Calcul de l'erreur

Calculer l'erreur à la dernière couche (L) :

$$\delta^{[L]} = \frac{\partial \mathcal{L}}{\partial a^{[L]}} \odot f'^{[L]}(z^{[L]})$$

où \mathcal{L} est la fonction de perte, et \odot représente le produit d'Hadamard (élément par élément).

3. Rétropropagation de l'erreur

Pour chaque couche $l = L - 1, \dots, 1$, rétropropager l'erreur :

$$\delta^{[l]} = ((W^{[l+1]})^T \delta^{[l+1]}) \odot f'^{[l]}(z^{[l]})$$

4. Mise à jour des poids et des biais

Mettre à jour les poids et les biais en utilisant le gradient de la fonction de perte par rapport à ces paramètres :

$$W^{[l]} = W^{[l]} - \eta \frac{\partial \mathcal{L}}{\partial W^{[l]}} = W^{[l]} - \eta \delta^{[l]} (a^{[l-1]})^T$$

$$b^{[l]} = b^{[l]} - \eta \frac{\partial \mathcal{L}}{\partial b^{[l]}} = b^{[l]} - \eta \delta^{[l]}$$

où η est le taux d'apprentissage.

Remarque : $f'^{[l]}(z^{[l]})$ représente la dérivée de la fonction d'activation par rapport à l'entrée $z^{[l]}$ de la couche l , qui est nécessaire pour le calcul des gradients par la règle de la chaîne.

Cette procédure est répétée pour un nombre défini d'itérations ou jusqu'à ce que la fonction de perte converge vers un minimum.

4.1.2 Avancées

4.1.2.1 Réseaux profonds

Avec le perfectionnement de la technique et l'augmentation exponentielle de la capacité de calcul, les réseaux de neurones ont pu croître pour devenir toujours plus profonds (avec de plus en plus de couches cachées). Supportant cette croissance, le perfectionnement de différents aspects de la technique. Que ce soit, les "skip connections (He *et al.*, 2016) facilitant la propagation du gradient sur de plus grandes distances, le "dropout" pour améliorer la robustesse et limiter le sur-apprentissage ou l'introduction des fonctions d'activation de type "ReLU" et "LeakyReLU", les avancées ont été nombreuses depuis la conception de l'algorithme original permettant aux modèles de l'état de l'art de passer de 8 couches avec le *AlexNet* (Krizhevsky *et al.*, 2012) à plus de 1000 couches pour les *ResidualNets* (He *et al.*, 2016).

4.1.2.2 Réseaux de neurones convolutifs

L'algorithme des réseaux de neurones convolutifs ou CNN (de l'anglais *convolutional neural networks*) est composée de plusieurs étapes clés, dont la convolution, l'activation, le pooling, et finalement la rétropropagation pour ajuster les poids. Formellement, les équations de base pour ces étapes sont:

1. Convolution :

$$Z_{ij}^{(l)} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} W_{mn}^{(l)} X_{(i+m)(j+n)}^{(l-1)} + b^{(l)} \quad (4.1)$$

où $Z_{ij}^{(l)}$ est la sortie de la couche de convolution l à la position (i, j) , $W_{mn}^{(l)}$ est le poids du noyau de convolution à la position (m, n) dans la couche l , $X_{(i+m)(j+n)}^{(l-1)}$ est l'entrée à la position $(i + m, j + n)$ de la couche précédente $l - 1$, et $b^{(l)}$ est le biais de la couche l .

2. Activation (par exemple, fonction ReLU) :

$$A_{ij}^{(l)} = \max(0, Z_{ij}^{(l)}) \quad (4.2)$$

où $A_{ij}^{(l)}$ est la sortie activée à la position (i, j) dans la couche l .

3. Pooling (par exemple, max pooling) :

$$P_{ij}^{(l)} = \max_{(m,n) \in R} A_{(i+m)(j+n)}^{(l)} \quad (4.3)$$

où $P_{ij}^{(l)}$ est la sortie du pooling à la position (i, j) dans la couche l , et R est la région du pooling autour de la position (i, j) .

4. **Rétropropagation** : La rétropropagation dans les CNN est plus complexe en raison de la convolution et du pooling. Le gradient de l'erreur par rapport aux poids de convolution est calculé en tenant compte de la contribution de chaque poids au signal de sortie. La formule générale est similaire à celle des réseaux de neurones pleinement connectés, mais l'application pratique nécessite de prendre en compte la structure spécifique des CNN. Sa formulation n'est pas particulièrement pertinente pour notre recherche et c'est pourquoi nous nous limitons aux principes généraux exposés ci-haut.

4.1.2.3 Transformeurs

Parmi les récentes avancées en recherche qui ont permises de faire augmenter les capacités techniques des réseaux de neurones artificiels, l'une d'entre elles se distingue particulièrement en matière de capacité et d'impact sur la recherche. Il s'agit des Transformeurs (Vaswani *et al.*, 2017). Les transformeurs ont été originalement conçus comme un mécanisme d'attention afin de permettre à un réseau plus traditionnel de stocker des informations en nombre beaucoup plus important que les techniques de l'époque, telles que les LSTM (Hochreiter et Schmidhuber, 1997), ne le permettaient, de façon plus durable, et plus rapidement (en nombre d'exemples) (Vaswani *et al.*, 2017). Les modèles de traitement du langage naturel les plus développés et les plus performants appartiennent d'ailleurs à cette famille d'architecture de réseaux: GPT-3 (Brown *et al.*, 2020), GPT-4 (OpenAI *et al.*, 2024), BERT (Devlin *et al.*, 2019).

Malgré l'approche mathématique qui a menée à leur élaboration (Vaswani *et al.*, 2017), ils sont en mesure de répliquer les représentations spatiales que l'on retrouve dans l'hippocampe (Whittington *et al.*, 2022) tel que les "neurones de grille" (Hafting *et al.*, 2005) et les "neurones de lieu" (O'Keefe, 1976). Cette architecture présente aussi de fortes similarités avec certains modèles de l'hippocampe développés en neurosciences, ce qui l'a initialement établie comme candidate d'algorithme de base pour notre recherche.

L'article (Ramsauer *et al.*, 2021) prouve que les modifications apportées à l'architecture des réseaux de neurones profonds pour obtenir un transformeur, font de ces derniers un isomorphisme computationnel (au niveau du traitement de l'information) aux réseaux de Hopfields continus modernes. Ces derniers sont fortement inspirés du principe d'auto-associativité observé dans l'hippocampe (Krotov et Hopfield, 2016). Cependant, puisque la version de la mémoire associative visée est celle des Hopfields modernes (Krotov et Hopfield, 2016), l'adaptation à partir des Hopfields classiques requise pour traiter les \mathbb{R} nécessite le retrait de la fonction Hebbienne. Cela l'éloigne du processus biologique analogue, raison qui nous a poussés à écarter cette architecture comme point de départ pour nos travaux.

Néanmoins il s'agit ici d'une démonstration comment l'intégration de principes computationnels et dynamiques du cerveau peut bénéficier aux algorithmes d'apprentissage modernes. C'est d'ailleurs, alliée aux performances impressionnantes des transformeurs, l'une des motivations qui nous a poussé à étudier les *Réseaux compétitifs Hebbiens*.

4.1.3 Limitations

4.1.3.1 Apprentissage lent

La propagation arrière, bien qu'efficace pour ajuster les poids synaptiques dans les réseaux de neurones profonds, souffre d'une convergence lente. Plus le réseau est profond et complexe, plus ce problème est magnifié, culminant avec des frais de calcul estimés à 25 millions \$ et 3 mois pour l'entraînement des dernières itérations des modèles de langage modernes. En plus de réduire l'accessibilité à l'entraînement, ce type de modèle à des compagnies disposant de fonds de recherche faramineux, ces coûts prohibitifs sont également des freins majeurs à l'avancement de la recherche puisque très peu de chercheurs ont accès à des ressources à cette échelle.

4.1.3.2 Méthodes séquentielles ad hoc

La nature "stateless" (sans état) de la propagation arrière, où chaque mise à jour des poids est effectuée indépendamment des états précédents du réseau, contraste fortement avec le fonctionnement dynamique et adaptatif du cerveau humain (Palva *et al.*, 2010; Knecht *et al.*, 1998). Ce dernier maintient et utilise des états internes persistants pour traiter les informations temporelles et contextuelles (Hammond, 2015; Tabas et von Kriegstein, 2021), une capacité que la propagation arrière standard ne capture pas. C'est grâce en

partie à ces états que les réseaux de neurones biologiques sont en mesure d'intégrer des informations temporelles (Rolls et Treves, 2011).

La rétropropagation ne disposant pas de mécanisme analogue, elle ne permet pas par défaut d'approximer des fonctions qui évoluent dans le temps et est limitée au traitement d'entrées statiques. De nombreuses techniques ont été développées au fil des années pour contrevenir à cette limitation. Les réseaux récurrents (Graves *et al.*, 2013) d'abord, mais qui nécessite de calculer tous les états intermédiaires du début à la fin de la séquence à apprendre, ce qui les rend gourmands en calcul et souffre des problèmes de la disparition et de l'explosion des gradients, réduisant son efficacité ainsi que son potentiel applicatif. L'architecture légèrement plus évoluée des LSTM (Hochreiter et Schmidhuber, 1997) vise à pallier au problème et bien qu'elle permette d'améliorer largement l'efficacité des modèles, cela n'élimine pas le problème de coût computationnel ainsi que l'impossibilité d'intégrer des informations qui précèdent la fenêtre de contexte établie lors de l'entraînement. Plus récemment, les architectures incluant le principe de l'attention (Vaswani *et al.*, 2017) ont permis de progresser significativement au niveau des séries séquentielles notamment dans le cas des modèles de langage massifs (LLM) qui ont désormais des capacités suffisamment avancées pour permettre leur application à une large gamme de tâches variées (OpenAI *et al.*, 2024).

4.1.3.3 Demande de calcul fixe

Toujours au niveau des limitations computationnelles, la rétropropagation, dans sa forme standard, requiert une quantité de calcul prédéterminée pour chaque passage avant et arrière à travers le réseau, indépendamment de la complexité intrinsèque ou de la simplicité de la tâche en cours d'apprentissage. Cette approche fixe ne reflète pas la capacité adaptative du cerveau humain à allouer des ressources de calcul de manière flexible, augmentant l'effort pour des tâches complexes tout en économisant de l'énergie pour des processus plus simples. Cette rigidité augmente dans un premier temps les coûts liés à l'exploitation de ce type de modèle puisque pour un nombre de mots fixé, une réponse à une question simple requiert le même calcul qu'une marginalement plus complexe. En second lieu, cela limite également les performances, car l'absence de cette capacité d'adapter la complexité du traitement requis en fonction de la complexité de la tâche fait en sorte qu'il n'est pas possible de dédier plus de ressources pour résoudre des problèmes qui requièrent des abstractions de plus haut niveau.

4.1.3.4 Difficulté de l'apprentissage par renforcement

La prémisse entière de l'algorithme de la propagation arrière est basée sur l'accès à une fonction d'erreur pouvant être appliqué sur la sortie de la dernière couche. Cela pose problème dans le cadre de l'apprentissage par renforcement, car cela signifie que nous devons définir une fonction de perte sur mesure pour le problème visé. Cela limite entre autres la généralisation d'une architecture à différents types de problématiques puisque cette fonction doit être redéfinie pour chaque application. Également, il est difficile d'évaluer la qualité d'une fonction de perte avant l'entraînement, ce qui est problématique, car celle-ci définit les performances qu'il sera possible d'obtenir avec un modèle donné.

4.1.3.5 Apprentissage non-supervisé

Comme pour le point précédent, le besoin d'une fonction d'erreur externe donnée pour l'entraînement réduit drastiquement son applicabilité dans un contexte d'apprentissage non supervisé. Bien que les travaux de (Devlin *et al.*, 2019) adaptent la rétropropagation pour effectuer des tâches non-supervisées, il s'agit de mécanismes ad hoc qui viennent eux-mêmes avec leur propre ensemble de limitations.

4.2 Apprentissage Hebbien

L'apprentissage Hebbien, nommé d'après Donald Hebb qui l'a initialement proposé dans son ouvrage "The Organization of Behavior" en 1949 (Hebb, 1949), est l'un des principes fondateurs en neuroscience et en intelligence artificielle qui stipule que le renforcement des connexions entre deux neurones est dirigé par les activations simultanées répétées de ceux-ci.

L'idée principale de l'apprentissage Hebbien peut être résumée par la phrase souvent citée: "Les neurones qui s'activent ensemble, se connectent ensemble." Mathématiquement, cela se traduit par une mise à jour des poids synaptiques selon la formule :

$$\Delta w_{ij} = \alpha * (x_i * x_j) \quad (4.4)$$

où w_{ij} est le poids de la connexion du neurone i au neurone j , α est le taux d'apprentissage, et $x_i, x_j \in \{0, 1\}$ sont les activations des neurones i et j , respectivement.

L'apprentissage anti-hebbien, au contraire, est la réduction du poids lorsque les neurones x_i et x_j sont inversement corrélés. Pour l'inclure dans l'équation 4.4 il suffit de modifier les valeurs d'activation tel que $x_i, x_j \in \{-1, 1\}$.

4.2.1 Règle d'apprentissage locale

Cette règle d'apprentissage est dite "locale" car la mise à jour du poids synaptique dépend uniquement des activités locales des neurones connectés par cette synapse, sans nécessiter une connaissance globale de l'état du réseau ou de l'erreur de sortie. Bien que l'apprentissage Hebbien remplisse cette contrainte, de nombreuses implémentations incluent des paramètres d'apprentissage globaux, tels que le taux d'apprentissage α , peuvent influencer l'échelle à laquelle les poids synaptiques sont ajustés. Par ailleurs, des mécanismes faisant fi de la localité comme la normalisation des poids ou la régulation homéostatique (Bush *et al.*, 2010) sont souvent considérés pour maintenir la stabilité globale du réseau et il peut être argumenté que ceux-ci ne contreviennent pas à la bio-plausibilité, ayant des analogues bio-chimiques (Lee et Kirkwood, 2019).

4.2.2 Apprentissage par renforcement

L'apprentissage Hebbien a été lié à l'apprentissage par renforcement, notamment par le biais de modèles qui intègrent une fonction de récompense comme facteur modulateur des changements synaptiques, favorisant un apprentissage orienté vers des objectifs spécifiques. Un exemple concret qui démontre que cette approche peut être utilisée en pratique pour transformer des influx neuronaux en signaux de contrôle pour un robot est présenté dans les travaux de (Mahmoudi *et al.*, 2013) qui sont parvenus à faire contrôler une prothèse par un sujet à l'aide d'un implant neuronal.

4.2.3 Temporalité

La dimension temporelle peut être intégrée dans l'apprentissage Hebbien (Mahmoudi *et al.*, 2013) pour prendre en compte le timing précis des activations neuronales, menant à des modifications comme la STDP, où l'ordre et le timing des potentiels d'action influencent la manière dont les connexions se renforcent ou s'affaiblissent. Cette variante sera visitée plus en détail dans la section 4.2.4.

4.2.4 Avancées

À travers les années, de nombreuses propositions tentant d'adapter l'algorithme de base à diverses fins ont vu le jour. Celles-ci vont de l'intégration de contraintes biologiques plus réalistes (Hopfield, 1982; Hopfield, 1984; Song *et al.*, 2000), au développement de modèles capables de simuler des fonctions cognitives complexes pour la recherche en neurosciences (Tomasello *et al.*, 2018; Hoerzer *et al.*, 2014; Bush *et al.*, 2010), en passant par l'application de principes Hebbiens pour l'apprentissage non supervisé de réseaux de neurones artificiels (Krotov et Hopfield, 2019) et pour l'apprentissage par renforcement tel que mentionné plus haut (Mahmoudi *et al.*, 2013).

4.2.5 Limites

Malgré son importance historique, l'apprentissage Hebbien présente des limitations, notamment sa tendance à conduire à une saturation, lorsque les poids sont limités à une valeur plafond, ou à une explosion, lorsque les poids ne sont pas limités, des poids synaptiques sans mécanismes de régulation¹. Aussi, bien que ses performances soient prometteuses dans certaines applications, à ce jour nous ne sommes toujours pas parvenus à en faire une approche aussi versatile et performante que la propagation arrière. Ces limitations ne sont cependant pas inhérentes à l'apprentissage Hebbien en soi, l'existence même du cerveau et ses nombreuses capacités décrites précédemment le prouvent, mais plutôt des effets collatéraux découlant des spécificités et des simplifications nécessaires à sa modélisation dans les implémentations historiques. Il inspire d'ailleurs des recherches continues pour développer des modèles d'apprentissage plus flexibles et adaptatifs, dont celles mentionnées dans la section précédente en sont quelques exemples.

4.3 Mémoires associatives (Hopfield)

Les *Hopfields* sont un ensemble de techniques d'apprentissage Hebbien qui sont basées sur l'approche proposée par (Hopfield, 1982). Il s'agit d'un réseaux Hebbien, dont les corrélations synaptiques entre deux neurones mènent au renforcement de leurs connexions, qui agit comme une mémoire adressable par contenu avec une activation binaire.

¹ Cela est facilement dérivable à partir de l'équation 4.4. Si un motif spécifique est représenté de nombreuses fois dans les données d'entraînement, les modifications successives s'additionnent rapidement pour mener à la saturation ou à une croissance rapides des poids impliqués dans ce motif.

4.3.1 Algorithme original

Lors de la phase d'apprentissage, le réseau est présenté avec un motif, à l'aide duquel il met à jour ses poids en suivant l'équation Hebbienne présentée dans la section précédente, soit:

$$\Delta W_{ij} = \alpha * (x_i * x_j) \quad (4.5)$$

Où x est le vecteur d'entrée (le motif), W est la matrice de poids et α est le taux (vitesse) d'apprentissage.

Ensuite, au moment de récupérer une information, on présente un motif corrompu correspondant à l'un de ceux sur lequel le modèle a été entraîné, et le réseau évalue en retournant le motif le plus près selon ses apprentissages.

La règle de mise à jour est la suivante

$$\hat{Y} = \text{sign}(W^T X) \quad (4.6)$$

Où \hat{Y} représente l'état interne du réseau (de chacun des neurones) ainsi que le vecteur de sortie.

La mise à jour de l'état peut être faite de façon simultanée comme dans 4.5, ou itérative (un neurone à la fois). Plusieurs itérations (mises à jour) successives peuvent être faites afin d'assurer que la sortie est dans un état stable et donc qui a convergé (pour lequel des mises à jour subséquentes ne produisent plus aucun changement). Pour notre approche présentée au chapitre 5, nous avons choisi d'appliquer la mise à jour synchrone de tous les neurones en raison de sa plus grande simplicité mathématique et computationnelle.

4.3.2 Avancées

4.3.2.1 Augmentation de la taille de la mémoire

Il est à noter que des états parasites émergent lorsque les motifs ne sont pas suffisamment uniques (orthogonaux) les uns par rapport aux autres ou lorsque nous excédons la capacité de stockage, qui est de 0.14 motif par neurone dans le modèle pour les *Hopfields* classiques (Hertz *et al.*, 1991). Cette capacité a été largement améliorée depuis (Storkey, 1997; Feng et Tirozzi, 1997; Gosti *et al.*, 2019) et varie grandement en fonction des différentes approches, chacune comportant des avantages et inconvénients qui leur sont propres mais dont les spécificités dépassent la portée de notre synthèse.

Plusieurs de ces techniques consistent à élargir le bassin d'attraction des motifs mémorisés, soit de maximiser la distance qu'il est possible d'avoir pour que le réseau converge entre le vecteur mémorisé et le vecteur présenté, afin de limiter la probabilité que des états parasites apparaissent (Folli *et al.*, 2017) (Storkey, 1999).

La méthode que nous proposons se distingue de ces approches par le fait qu'elle vise à effectuer une classification et non une simple mémorisation. Notre but est par conséquent, à l'instar de cette technique, d'élargir les bassins d'attractions des attracteurs associés à chacun des motifs, qui sont des chiffres dans notre implémentation, afin que celui-ci englobe toutes les instances possibles du même chiffre (0-9).

4.3.2.2 Hopfield continus

Les Hopfields ont également des variantes qui travaillent sur des nombres \mathbb{R} en place de simples motifs binaires (Hopfield, 1984; Movellan, 1991; Talaván et Yáñez, 2005). Cependant leur plus grande complexité rend leur adaptation plus ardue à notre cas d'utilisation et c'est pourquoi nous nous sommes simplement basés sur la version binaire pour notre recherche.

4.3.2.3 Mémoire associative bi-directionnelle

On voit à travers la formule 4.5 que la matrice de poids obtenue est symétrique puisqu'il n'y a pas de distinction entre l'entrée et la sortie. Dans cette forme simple, la fonction de mise à jour est de type Lyapunov (Hertz *et al.*, 1991) ce qui garanti qu'elle convergera ultimement vers un état d'énergie qui est un minima d'énergie (telle que définie dans (Hopfield, 1982)) local (Liapounoff, 1907) (ou une oscillation entre deux états lorsque la mise à jour de tous les noeuds est simultanée). Une approche différente est d'associer un motif avec un autre, générant une mémoire associative bi-directionnelle (Kosko, 1988).

L'équation pour l'apprentissage d'une matrice de poids qui encode une série d'associations $(X_0, Y_0), \dots, (X_n, Y_n)$ est simplement:

$$\Delta W = \sum_{i=0}^n X_i^T Y_i \quad (4.7)$$

Pour ce qui est de récupérer Y_i à partir de X_i , il s'agit de la même formule que pour les Hopfields standards,

telle que définie dans 4.6.

Cette approche est intéressante pour notre recherche puisqu'en place d'un attracteur sur un point fixe, elle permet d'encoder une série d'états successifs, à l'image du fonctionnement de l'hippocampe, permettant l'encodage de séries temporelles. Également, il est possible d'encoder des cycles simplement en apprenant une série d'associations du type $(X_0, X_1), (X_1, X_2), \dots, (X_n, X_0)$. De cette façon, X_0 génère X_1 , X_1 génère X_2 et ainsi de suite jusqu'à ce que la boucle soit bouclée avec X_n qui génère le point de départ du cycle.

4.3.3 Limitations

Il existe des limitations majeures qui freinent l'application industrielle de ces techniques. En effet, malgré les améliorations qu'elles ont subies et leur prévalence, avec plus de 27 000 citations enregistrées, dans la littérature, les Hopfields ne sont toujours pas utilisés de façon majeure dans aucun domaine à l'extérieur de la recherche.

4.3.3.1 Techniques plus efficaces

Les mémoires associatives démontrent des capacités intéressantes qui ont des applications telles que le stockage d'informations, la compression, la correction d'erreur, trouver le plus proche voisin d'un vecteur dans l'ensemble des vecteurs mémorisés et, pour les mémoires associatives bi-directionnelles, d'associer une entrée avec une sortie. Cependant, pour chacun ces cas d'applications, il existe des techniques qui sont plus optimisées autant au niveau de l'espace de stockage que de la demande en calcul requis, ce qui rend leur utilisation non souhaitable pour les cas où l'on tente d'effectuer l'une de ces opérations. Là où elles se démarquent est leurs capacités d'effectuer toutes ces tâches à la fois. C'est précisément la raison pour laquelle nous nous y intéressons dans le cadre de notre recherche.

4.3.3.2 États parasites

Comme mentionné plus haut, une mémoire associative peut converger vers un état dit "parasitaire" qui est une chimère entre deux entrées et qui représente un minima local dans la mise à jour des états. Ces états chimères qui sont généralement une combinaison entre deux motifs mémorisés (Hertz *et al.*, 1991) freinent l'adoption de cette méthode puisqu'il n'existe pas de méthode directe et peu coûteuse en termes

de calcul pour déterminer si l'état final représente bien un vecteur qui a été sauvegardé dans le réseau de façon volontaire ou involontaire.

4.4 Apprentissage compétitif

L'apprentissage compétitif est une sous-catégorie de l'apprentissage dans les réseaux de neurones, qui consiste à attribuer une fonction d'utilité à chaque neurone et à effectuer un apprentissage modulé par cette valeur sur les données entrantes.

4.4.1 Cartes de Kohonen

La technique la plus populaire qui l'incorpore est celle des cartes de Kohonen (Kohonen, 1990), aussi appelée dans la littérature "cartes autoadaptatives" (en anglais, Self-Organizing Maps ou SOM). Le principe consiste à sélectionner un neurone gagnant pour chacun des intrants, et d'attribuer les poids permettant l'activation maximale de ce neurone pour les prochaines itérations ou cette donnée sera présentée. Lorsque les poids d'un neurone gagnant sont mis à jour, on met également à jour, dans une mesure moindre, les poids des neurones qui lui sont environnants, comme déterminé par une fonction de proximité entre les neurones. De cette façon ceux-ci représentent une couverture de l'espace des intrants possibles.

4.4.1.1 Algorithme

La mise à jour des poids dans un SOM est formulée de la manière suivante:

$$\Delta w_{ij} = \alpha(t) \cdot h_{ci}(t) \cdot (x - w_{ij})$$

où : - w_{ij} est le poids entre l'élément i du motif d'entrée et le neurone j dans la carte. - $\alpha(t)$ est le taux d'apprentissage à l'instant t , qui diminue généralement avec le temps. - $h_{ci}(t)$ est la fonction de voisinage autour du neurone gagnant c à l'instant t , qui détermine l'influence du neurone gagnant sur ses voisins. - x est le vecteur d'entrée.

La fonction de voisinage $h_{ci}(t)$ comme la fonction gaussienne est souvent choisie:

$$h_{ci}(t) = \exp \left(-\frac{\|r_c - r_i\|^2}{2\sigma^2(t)} \right)$$

où : - r_c et r_i sont les positions des neurones c (gagnant) et i dans la carte, respectivement. - $\sigma(t)$ est l'écart-type de la fonction de voisinage à l'instant t , qui diminue également avec le temps.

4.4.1.2 Avancées

Depuis leur création, les cartes de Kohonen ont connu plusieurs avancées, modifications et améliorations significatives, élargissant ainsi leur champ d'application et leur efficacité. L'une des évolutions notables est l'introduction de techniques d'apprentissage adaptatif, qui permettent de régler dynamiquement les paramètres tels que le taux d'apprentissage et le rayon de voisinage en fonction de l'état d'apprentissage du réseau (Shah-Hosseini et Safabakhsh, 2000). Cela conduit à une convergence plus rapide et à une meilleure qualité de la cartographie. Les variantes dynamiques des cartes de Kohonen, capables de modifier leur structure, tel qu'avec l'ajout de neurones au cours de l'apprentissage, ont été développées pour mieux s'adapter à la complexité des données traitées (Hsu *et al.*, 2009). L'intégration de la notion de temps, avec des modèles tels que les cartes de Kohonen récurrentes, a permis le traitement de séquences temporelles et de données séquentielles (Shah-Hosseini et Safabakhsh, 2000). En outre, l'application des cartes de Kohonen à des domaines spécifiques a entraîné le développement de variantes spécialisées, comme celles conçues pour le traitement du langage naturel (Honkela, 1998) ou pour des applications biomédicales (Skupin *et al.*, 2013), où elles aident à identifier des modèles complexes dans les données génétiques ou protéomiques. Ces avancées ont non seulement enrichi la théorie sous-jacente des cartes de Kohonen mais ont également étendu leur portée à de nouvelles frontières de la recherche et de leur application pratique. Cependant, malgré ces améliorations, elles offrent rarement les meilleures performances ce qui explique que leur utilisation pratique demeure limitée dans l'industrie.

4.4.1.3 Limites

Mis à part les applications pointues où elle excelle, cette approche, demeure relativement ésotérique et s'applique généralement à des problèmes bien spécifiques en raison des limitations inhérentes à sa structure. Tout d'abord, la convergence du modèle n'est pas toujours garantie (Kohonen, 2013), et les résultats peuvent varier significativement en fonction de l'initialisation et de l'ordre de présentation des données. Enfin, leur nature statique ne permet pas de traiter efficacement des données séquentielles de

façon innée, ce qui réduit leur applicabilité à des tâches où la dimension temporelle est importante.

4.5 Attracteurs spatio-temporel hebbien

Molter et Bersini ont travaillé sur une série de recherches (Molter *et al.*, 2006) portant sur les attracteurs spatio-temporaux dans des réseaux hebbien récurrents, se distinguant de toutes les techniques présentées jusqu'ici qui portent sur des attracteurs sur point fixe.

Dans leurs travaux, ils introduisent la notion de frustration dans un réseau récurrent Hebbien, qui renvoie à une dynamique où le réseau neuronal se trouve dans un état de chaos frustré: "Le chaos frustré est un régime dynamique qui apparaît dans un réseau lorsque la structure globale est telle que les motifs de connectivité locaux, responsables de comportements oscillatoires stables et significatifs, sont entrelacés, menant à des attracteurs en compétition mutuelle et à une itinérance imprévisible parmi des apparitions brèves de ces attracteurs."(Molter *et al.*, 2006)². À travers leurs expériences, ils sont parvenus à démontrer que ce régime dynamique permettait la génération de dynamiques complexes, incluant des régimes chaotiques, qui sont utiles pour la prévention de la prolifération de données parasites lorsque le réseau est confronté à des stimuli ambigus ou non appris.

La frustration déstabilise le réseau et provoque un 'vagabondage' erratique parmi les orbites périodiques caractérisant le même réseau lorsqu'il est connecté de manière non frustrée . Celle-ci est obtenue par la superposition de plusieurs matrices de connectivité qui, prises individuellement, conduiraient à des régimes oscillatoires, induit un chaos qui est caractérisé par la détection d'orbites répulsives cachées dans l'attracteur chaotique.

4.5.1 Algorithme

Le RNN qui est présenté dans (Molter *et al.*, 2006) et exhibant des dynamiques de chaos frustré vise à traiter des stimuli externes pour générer des dynamiques internes qui peuvent apprendre et stocker des informations sous forme d'attracteurs spatio-temporels est défini comme suit:

Le réseau est pleinement connecté, chaque activation neuronale étant une fonction des impacts des autres neurones ainsi que des stimuli externes. Les activations neuronales sont mises à jour de manière synchrone

² En anglais, traduction libre

à chaque pas de temps discret. La description mathématique du réseau est la suivante :

L'activation x_i du neurone i au temps $n + 1$ est donnée par :

$$x_i(n + 1) = f(\text{net}_i(n)) \quad (4.8)$$

où $\text{net}_i(n)$ est le potentiel d'entrée pour le neurone i au temps n , calculé comme :

$$\text{net}_i(n) = \sum_{j=1}^N w_{ij}x_j(n) + \sum_{s=1}^M w_{is}\iota_s \quad (4.9)$$

- N est le nombre de neurones dans le réseau.
- M est le nombre d'unités constituant le stimulus.
- w_{ij} est le poids synaptique de la connexion du neurone j au neurone i .
- w_{is} est le poids de l'unité s du stimulus externe sur le neurone i .
- ι_s est la s -ième unité du stimulus externe.
- f est une fonction d'activation saturante (ici \tanh), utilisée pour assurer la non-linéarité du réseau et limiter l'amplitude des activations neuronales.

4.5.1.1 Algorithme d'apprentissage

L'apprentissage dans ce réseau est réalisé à l'aide d'un mécanisme Hebbien asymétrique, qui est adapté pour capturer et renforcer des corrélations temporelles spécifiques entre les entrées et les sorties. Deux tâches d'apprentissage principales sont proposées:

1. *Apprentissage "out-supervised"* : Dans cette approche, des stimuli externes sont explicitement mappés à des attracteurs cycliques définis au préalable par les chercheurs. Les poids sont ajustés à l'aide d'une

fonction Hebbienne pour renforcer la stabilité de ces mappings.

2. *Apprentissage "in-supervised"* : Cette méthode est moins supervisée et biologiquement plausible. Elle permet au réseau de créer ses propres représentations internes des stimuli externes sans que des attracteurs spécifiques ne soient prédéfinis. Le réseau apprend à réagir aux stimuli en générant des séquences cycliques internes, dont la périodicité et la structure ne sont pas spécifiées à l'avance.

CHAPITRE 5

CONTRIBUTION

Maintenant que nous avons établi la base théorique sur laquelle nous nous appuyons pour formuler notre algorithme, nous pouvons formaliser celui-ci. Pour ce faire, nous tâchons d'abord de distiller notre problématique et les notions vues précédemment pour ensuite proposer notre propre modèle de réseau de neurones récurrent bio-plausible (bio-RNN) et son algorithme d'apprentissage.

5.1 Méthodologie

Nous avons établi dans notre revue de littérature que l'apprentissage par rétropropagation comporte des limitations qui sont inhérentes à sa formalisation. Bien que l'avancement de ces techniques semble accélérer de façon exponentielle à ce moment, nous croyons que ses déficiences vont ultimement définir un seuil au-delà duquel il sera coûteux de maintenir ce rythme de progression. Alors que l'originalité de la propagation arrière s'estompe peu à peu et que les limites qui en découlent deviennent plus apparentes, l'intérêt porté sur les techniques alternatives augmente. Comme par le passé, beaucoup se tournent vers les neurosciences pour y trouver de l'inspiration et des intuitions. Que ce soit le besoin d'une fonction d'erreur prédéfinie, le nombre faramineux d'exemples nécessaires pour atteindre une saturation de l'apprentissage (le moment qui précède immédiatement le sur-apprentissage), le manque d'une dimensionnalité temporelle, l'oubli catastrophique ou d'autres facteurs, l'absence de ces contraintes dans le cerveau humain est une preuve que ces limitations ne sont pas inhérentes à l'apprentissage dans des réseaux de neurones profonds. Ce ne sont que des restrictions engendrées par nos techniques actuelles.

Nous avons choisi d'explorer les bio-RNN pour de multiples raisons. (1) Nous croyons que certaines des avancées qui ont rendu possible l'évolution rapide et les gains de performance des RN avec RP pourraient être applicables et ainsi bénéficier à de nombreux algorithmes développés par le passé. L'augmentation de la disponibilité des jeux de données toujours plus larges, l'augmentation de la capacité de calcul et les optimisations mathématiques de l'apprentissage sont des exemples de facteurs qui peuvent potentiellement profiter à ces techniques aujourd'hui. (2) La progression constante de la compréhension des mécanismes neurologiques sous-jacents à la cognition est également un atout qui peut nous inspirer pour le développement de nos algorithmes ainsi que nous renseigner sur les pistes d'optimisation qui semblent être les plus prometteuses. (3) Nous croyons qu'il existe de meilleures formalisations algorithmiques que la RP pour l'apprentissage dans les RN, ouvrant la porte à des découvertes majeures améliorant substantiellement les performances sur une large gamme de tâches. (4) Finalement, nous croyons que les algorithmes bio-plausibles, en raison des éléments qui vont dans ce sens que nous avons présentés, sont la voie avec le plus de potentiel pour

faire avancer l'IA vers l'intelligence artificielle générale.

5.1.1 Cadre expérimental

Pour la présente recherche, nous nous intéressons à un problème relativement simple afin d'être en mesure d'évaluer notre algorithme malgré son manque de maturité et l'absence d'optimisations qui en découle. Nous nous sommes arrêtés sur la classification d'image comme tâche d'entraînement et d'évaluation de notre modèle. Ce choix est motivé dans un premier temps par la riche histoire de la recherche sur cette problématique. Cela implique qu'il y a une grande variété dans les jeux de données disponibles, autant en terme de taille que de complexité. L'existence de jeux de données simple nous permet d'évaluer notre algorithme malgré son manque de maturité et d'itérer rapidement en raison du temps d'entraînement réduit que ceux-ci requièrent pour que l'apprentissage soit observable. L'existence de jeux de données plus complexe fait en sorte que nous pourrions aisément augmenter le niveau de complexité dans des recherches futures. Dans un second, la littérature abondante sur ce sujet et les implémentations diversifiées qu'elle nous offre nous permet d'évaluer adéquatement où se situe notre algorithme en rapport avec les techniques existantes.

5.1.1.1 Le jeu de données MNIST

Pour l'entraînement et l'évaluation de notre algorithme, nous avons retenu le populaire jeu de données MNIST (Deng, 2012). Ce jeu de données est fréquemment utilisé pour tester de nouveaux algorithmes en raison qu'il est (1) suffisamment simple pour un algorithme non-optimisé d'être en mesure d'obtenir un score raisonnable dès lors qu'il a une capacité d'apprentissage non-nulle, (2) suffisamment complexe pour démontrer que l'algorithme est en mesure d'apprendre et/ou de mémoriser des caractéristiques latentes non-triviales dans le jeu de données et (3) son utilisation fréquente dans la littérature nous permet également de comparer directement notre algorithme avec d'autres techniques (Deng, 2012; LeCun et Cortes, 2005; Krotov et Hopfield, 2019; Byerly *et al.*, 2020) et simplifie le processus d'implémentation de modèles comparatifs. Ces éléments sont d'autant plus applicables dans la recherche présentée ici puisqu'une des métriques à laquelle nous accordons de l'importance est la vitesse d'apprentissage au moment de l'initialisation et non la capacité théorique après un nombre élevé d'itérations sur l'ensemble d'entraînement.

5.1.1.2 Protocole des données

Pour évaluer notre modèle, nous avons décidé d'utiliser un sous-ensemble du jeu de données MNIST d'une taille de 10000 exemples. Cet ensemble a été mis de côté et n'est pas utilisé lors de l'entraînement. La taille de jeux d'entraînement, quant à elle, est de 60000 images. Les images du jeu de données sont des images en noir et blanc, qui se présentent sous la forme de matrices de 28×28 (une par image) contenant des valeurs entre 0 et 256. Puisque notre réseau est binaire, nous transformons au préalable l'image en une image binaire. D'autres options s'offraient à nous, tel que mettre en entrée les valeurs telles quelles, normalisées entre 0 et 1, ou encore d'encoder temporellement l'intensité des pixels. Cependant, la gestion des dynamiques dans des systèmes relativement large comme le nôtre est déjà suffisamment ardue dans ses formes les plus simples, ce qui nous a poussés à sélectionner l'option la plus accessible nous permettant de mieux isoler d'autres variables essentielles lors de nos tests.

5.1.1.3 Modèles comparatoires

Afin de pouvoir situer les performances de notre algorithme par rapport aux techniques populaires actuelles, nous avons choisi d'implémenter quelques réseaux de neurones artificiels simples composés d'une et de deux couches cachées. Notre choix de sélectionner des architectures simples est motivé par plusieurs facteurs. Dans un premier temps, le nombre d'itérations d'apprentissage requises pour obtenir un résultat donné croît avec la largeur et la profondeur du réseau.

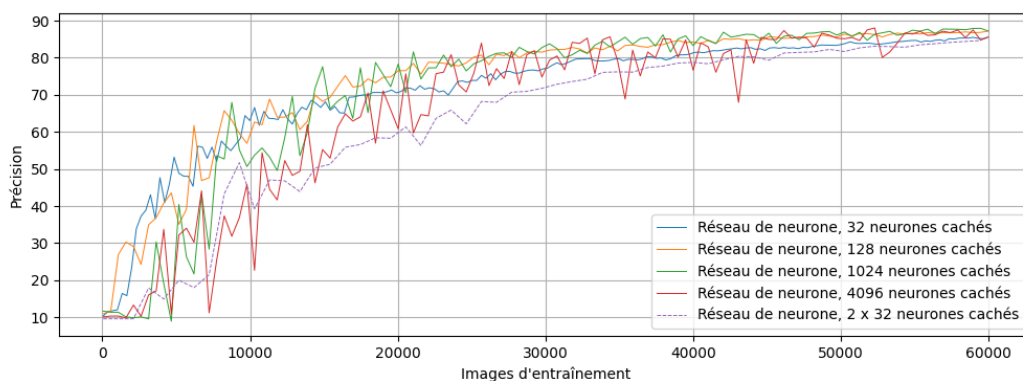


Figure 5.1 Évolution de l'apprentissage pour différentes tailles de RN

Puisque le cadre de notre recherche est de démontrer que notre approche est viable, nous jugeons que la démonstration et la comparaison de ses capacités à petite échelle est plus appropriée. La réduction en

complexité résultante nous permet de (1) itérer plus rapidement grâce à un temps d'entraînement réduit, (2) d'écarter ainsi plus rapidement les approches moins prometteuses, (3) de réduire le nombre de variables à investiguer avant de pouvoir déterminer si une technique ou modification est viable ou non, (4) de réduire le nombre d'hyper-paramètres à explorer et (5) de mesurer notre approche avec d'autres algorithmes ayant un niveau similaire de maturité. Les résultats obtenus sur le jeu de données avec les différentes architectures sont présentés dans la figure 5.1.

Ces résultats nous permettent de comparer notre approche avec la RP en évaluant l'impact des variations du nombre de neurones et du nombre d'exemples d'entraînement sur la précision du modèle. Il est à noter que nous avons utilisé la fonction d'activation sigmoïde pour obtenir les résultats présentés dans la figure 5.1. Nous avons fait ce choix dans le but de représenter la RP dans sa forme traditionnelle. Le tout, dans le but d'être en mesure de la comparer dans un contexte similaire à notre méthode, sans les nombreuses optimisations intégrées suite à près d'un demi siècle de recherche.

Nous souhaitons souligner deux points clés à retenir qui seront pertinents lors de la présentation de nos résultats à la section 5.3. Le premier est qu'avec l'augmentation du nombre de neurones et de couches, la vitesse de l'apprentissage semble diminuer progressivement en début d'entraînement. Ce facteur est résolu à la fin de l'exécution. Sur des entraînements plus longs que ceux illustrés dans 5.1, les modèles plus larges prennent inévitablement le dessus sur ceux plus petits, mais nous n'avons pas jugé pertinent de les inclure puisque nous mettons plus d'importance sur la rapidité d'apprentissage que sur la performance maximale. Le second est qu'avec l'augmentation de la taille du réseau, la précision devient de moins en moins stable.

5.1.2 Contraintes biologiques

À ce stade, un bon nombre de concepts théoriques multidisciplinaires qui peuvent être pris en compte pour l'élaboration d'un algorithme d'apprentissage pour des réseaux récurrents bio-plausibles (Bio-RNN) ont été couverts. Cependant, il ne nous est malheureusement pas possible de les intégrer dans leur ensemble dans la présente recherche en raison du niveau élevé de complexité que cela entraînerait. Autant au niveau de l'implémentation que de l'évaluation, un nombre croissant de paramètres rends l'étude de l'impact de chacun d'entre eux plus ardue. Nous avons donc choisi un sous-ensemble composé de ceux que nous croyons être les plus pertinents pour notre contexte expérimental afin de les implémenter.

Le terme "bio-plausible" peut être appliqué à une grande famille d'algorithmes avec une définition qui varie en fonction de du domaine et de la recherche en question, en raison du grand nombre de contraintes et de la complexité des processus variés d'apprentissage dans le cerveau. Nous avons donc posé notre propre sélection sur les éléments que nous considérons inclure dans notre approche. Au niveau de la bio-plausibilité nous avons retenu les contraintes décrites dans les paragraphes suivants.

5.1.2.1 Règles d'activation et de plasticité locales

L'une des contraintes que nous nous imposons est que la règle d'apprentissage soit locale. Concrètement, cela implique que toutes les informations qui mènent à une modulation du lien synaptique et de l'activation proviennent des neurones pré- et post-synaptiques. Concrètement, les neurones mettent à jour leurs poids simultanément et une connexion entre deux neurones ne tient compte que des informations présentes et historiques appartenant à son champ de perception (les neurones de la couche précédente, courante et suivante). Cette contrainte, nous croyons, ouvre la porte à une meilleure mise en échelle de notre proposition puisqu'elle permet un accroissement du nombre de neurones dans le réseau sans que cela n'ait un impact sur le fonctionnement local des neurones déjà présents dans une architecture donnée. Ce choix est également motivé par le fait qu'il permet une meilleure optimisation éventuelle s'il s'avérait que cet algorithme doive être parallélisé sur plusieurs cartes graphiques puisque les calculs de mise à jour ne dépendent que d'informations locales.

5.1.2.2 Fonction d'activation binaire

Nous avons choisi d'utiliser une fonction d'activation binaire en place d'une fonction non-linéaire continue. Bien que de l'information soit perdue dans la compression dichotomique, celle-ci offre différents avantages qui compensent la perte d'information accessible aux calculs subséquents. Puisque notre réseau n'utilise pas la rétropropagation, nous pouvons nous permettre d'utiliser une fonction non différentiable. Elle simplifie l'analyse numérique des dynamiques, l'implémentation de l'algorithme, et l'analyse visuelle des états et des modifications des poids synaptiques suite à l'entraînement. D'ailleurs une branche entière de l'intelligence artificielle est consacrée réseaux de neurones à pics (SNN, de l'anglais "spiking neural networks") en raison de leur efficacité computationnelle. Cette efficacité permet de concevoir des processeurs neuromorphiques optimisés spécialement pour les SNN (Lobo *et al.*, 2020) à faible demande énergétique, qui est l'une des motivations derrière ces recherches (Asghar *et al.*, 2021). L'utilisation de cette fonction nous permet donc à terme d'envisager exécuter notre algorithme sur de tels processeurs, ce qui représente un avantage par

rapport au côté énergivore de la RP.

5.1.2.3 Plasticité temporelle

Comme nous l'avons vu à la section 3.5.1, la STDP est l'un des mécanismes ayant attiré à la plasticité des plus étudiés et il est soupçonné qu'elle soit l'un des mécanismes fondamentaux de l'apprentissage biologique. Par conséquent il s'agit de l'un des principes qu'il nous semblait naturel d'intégrer à notre algorithme. Puisque nous visons à induire des cycles limites de période $N = 2$, comme spécifié à la section 5.2.1.1, pour valider notre approche avant de tenter de créer des attracteurs plus complexes, cela implique simplement d'appliquer une fonction Hebbienne et anti-Hebbienne qui prend en compte les activations des neurones aux temps $t - 1$, t et $t + 1$.

5.1.2.4 Inhibition équilibrée

Niu et al. ont démontré que dans un cadre de système dynamique, il est possible d'encoder une grande quantité d'informations avec très peu de neurones actifs (Niu *et al.*, 2022). Les recherches de Yu et al. suggèrent qu'un ratio $\text{excitation}/\alpha\text{inhibition}$ équilibré, avec α le facteur d'échelle approprié, permet des dynamiques saines permettant d'encoder un maximum d'information tout en permettant une économie des ressources énergétiques (Yu *et al.*, 2018). Ils proposent qu'il s'agit également d'une adaptation optimisant les dynamiques des influx nerveux pour atteindre une meilleure extraction des connaissances à partir de l'information sensorielle et une meilleure capacité de rétention de l'information. Le cerveau lui-même est peu densément connecté et à tout moment, seulement une petite fraction des neurones sont actifs simultanément (Wixted *et al.*, 2014). À ce stade il n'existe toujours pas de preuve qu'un cerveau ne pourrait être composé de moins de neurones et être plus densément connecté en offrant le même niveau d'entropie que leur forme actuelle et avec une dépense énergétique similaire. Cela nous pousse à croire qu'il y a donc des justifications fonctionnelles et computationnelles pour maintenir une activité globale relativement faible au niveau de l'impact de cette mesure sur les dynamiques. Pour ces raisons, nous jugeons judicieux d'inclure ce paramètre dans notre modélisation.

5.1.3 Attracteurs limites

La plupart des méthodes modernes d'IA, dont tout le domaine de l'apprentissage profond utilisant la propagation arrière, les SVM, les cartes de Kohonen et autres ne sont pas considérés comme étant des systèmes dynamiques

au moment de l'inférence. Ce sont simplement des fonctions paramétrées au moment de l'entraînement. Une fois la phase d'apprentissage terminée, ils convergeront toujours vers le même point fixe lorsque présenté avec une entrée donnée.

Le cerveau quant à lui est un système dynamique à états, où les états passés ont un impact sur le fonctionnement et le résultat des calculs. La dimension spatio-temporelle guide les dynamiques en dictant les états successifs du système. Cela lui permet entre autres de traiter des données séquentielles de façon innée, puisque cette dimension temporelle est nécessaire pour l'apparition de cycles-limites, qui ont eux-même une composante temporelle. Certaines méthodes comme les réseaux de Hopfields et leurs variantes (Hopfield, 1982; Hopfield, 1984; Krotov et Hopfield, 2019; Kosko, 1988; Storkey, 1997), en sont également, mais leur attracteur converge néanmoins vers un point fixe. Notre approche se démarque de la majorité des approches présentées, à l'image de (Molter *et al.*, 2006; Molter *et al.*, 2007) en tentant d'encoder le traitement de l'information dans un attracteur encodé à l'intérieur d'un cycle-limite plutôt que sur un point fixe.

De nombreux travaux démontrent que des attracteurs dynamiques sains sont nécessaires pour un traitement cérébral approprié de l'information (Preissl *et al.*, 1996; Tomasello *et al.*, 2018; Molter *et al.*, 2006; Faisal *et al.*, 2008; Molter *et al.*, 2007; Molter *et al.*, 2004). Cela suggère un avantage computationnel de cette approche par rapport à un attracteur sur un point fixe.

Les résultats de Molter *et al.* sur l'augmentation de la capacité de stockage des mémoires associatives lorsque des cycles-limites sont mémorisés en place d'entrées fixes (Molter *et al.*, 2004). Dans des travaux subséquents (Molter *et al.*, 2006; Molter *et al.*, 2007) sur le chaos frustré ainsi que sur l'apprentissage non supervisé dans leur modèle (Molter *et al.*, 2005) nous font croire qu'il s'agit d'une approche viable et prometteuse.

5.1.4 Inspiration algorithmique

5.1.4.1 Hopfield

L'algorithme de base derrière notre version n'est autre que celui des mémoires associatives. La variante sélectionnée est celle bi-directionnelle et où tous les neurones sont mis à jour de façon synchrone.

Dans le cas de la directionnalité, notre choix est motivé par le fait que nous voulons établir des attracteurs

limites, dans notre cas nous ciblons des cycles ayant une période de longueur N . Pour ce faire le réseau doit apprendre des transitions d'états convergeant sur un cycle limite et non vers un point fixe.

5.1.4.2 Chaos frustré

Nous avons également emprunté la notion de chaos frustré défini dans Bersini et Molter (Molter *et al.*, 2006; Molter *et al.*, 2007) et repris leur version de mémoire associative spatio-temporelle présentée dans (Molter *et al.*, 2005). La forme de la fonction d'activation reste similaire à l'équation 4.8 et nous généralisons l'équation 4.9 à n couches pour prendre en compte notre couche de sortie. Nous obtenons donc, pour la couche d'entrée $l = 0$:

$$\text{net}_i^0(n) = \iota_i \quad (5.1)$$

- ι_s est la j -ième unité du stimuli externe (le vecteur d'entrée).

Pour les couches cachées, nous avons:

$$\text{net}_i^l(n) = \sum_{j=1}^{N^{l-1}} w_{[ji]}^{[l-1,l]} \sigma(x_j^l(n-1)) + \sum_{j=1}^{n^l} w_{ji}^{l,l} \sigma(x_j^l(n-1)) + \sum_{j=1}^{n^{l+1}} w_{ji}^{[l+1,l]} \sigma(x_j^l(n-1)) \quad (5.2)$$

- l est le nombre de couches dans le réseau.

- n^l est le nombre de neurones dans la couche l du réseau.

- $w_{j,i}^{k,l}$ est le poids synaptique de la connexion du neurone j de la couche k au neurone i de la couche l .

- σ est une fonction d'activation saturante utilisée pour assurer la non-linéarité du réseau et limiter l'amplitude des activations neuronales. dans notre cas, celle décrite par l'équation 5.4.

Finalement pour la couche de sortie $l = L$ nous avons:

$$\text{net}_i^l(n) = \sum_{j=1}^{n^{l-1}} w_{[ji]}^{[l-1,l]} \sigma(x_j^l(n-1)) \quad (5.3)$$

cette dernière pourrait inclure des connexions récurrentes pour le calcul de son état, cependant dans notre cas nous travaillons avec des vecteurs de sortie de type "one-hot" (un seul des neurones est activé à la fois), ce qui rend ce calcul superflu.

l'objectif de notre approche est de formaliser une règle de mise à jour de l'état et des connexions d'un réseau récurrent qui sera en mesure d'apprendre à converger vers différents cycles limites déterminés par les données d'entrée démontrant la capacité d'effectuer un traitement de l'information. leurs travaux représentent un pas dans cette direction puisqu'ils ont démontré qu'il s'agissait d'une avenue viable pour mémoriser différents motifs. nous tenterons de démontrer que ça l'est également dans le cas du traitement de l'information, en posant comme hypothèse que celui-ci sera en mesure d'identifier des régularités statistiques provenant du jeu de données et d'en extraire des caractéristiques significatives pour effectuer une tâche de classification.

5.1.5 Notation

afin de simplifier les formules pour en faciliter la compréhension, nous utilisons la notation matricielle pour les équations qui suivent. les différentes variables qui seront principalement utilisées dans celles-ci sont:

w est l'ensemble des matrices de poids pour le réseau

$w^{a,b}$ est une matrice de poids (connexions) de la couche a à la couche b du réseau. $w^{a,a}$ est donc une matrice de connexions récurrente pour a . pour notre réseau simple avec une couche d'entrée, une couche cachée et une couche de sortie présenté ici, nous avons donc $w^{0,1}$ qui sont les poids connectant la couche d'entrée à la couche cachée, $w^{1,1}$ qui sont les connexions récurrentes de la couche cachée et $w^{1,2}$ qui sont les poids de la couche cachée à la couche de sortie.

$w_{i,j}^{a,b}$ est le poids entre le neurone i de la couche a et le neurone j de la couche b .

x est un tenseur contenant tous les états.

$x^l[t]$ est la matrice d'états de la couche l au temps t . dans un réseau à l couches, la couche $l = 0$ est la couche d'entrée, les couches $l = \{1, \dots, l - 2\}$ sont les couches cachées et la couche $l = l - 1$ est la couche de sortie.

$x_i^l[t]$ est l'états du neurone i de la couche l au temps t .

img est la valeur du vecteur d'entrée, soit les images du dataset.

p_{max} est la valeur minimum pour un poids.

p_{min} est la valeur maximum pour un poids.

m_{seuil} est la valeur au-delà de laquelle un neurone s'active.

m_{max} est la valeur maximum pour l'état d'un neurone.

m_{min} est la valeur minimum pour l'état d'un neurone.

e est la valeur d'énergie d'un neurone.

voici également la notation des opérateurs que nous utilisons

○ produit d'hadamard, ou par élément.

× produit matriciel, ou le produit cartésien dans le cas de multiplication de 2 vecteurs.

m^T transposée de m .

certaines fonctions sont construites à travers plusieurs équation, afin d'en simplifier la compréhension. dans ces cas, nous utilisons la notation prime. si nous tentions d'exprimer une fonction complexe $f(x)$ à l'aide de trois équations nous aurions:

$f'(x)$ serait la fonctions dans sa version la plus simple

$f''(x)$ est une fonction plus complexe (p. ex. on y a ajouté un terme)

$f(x)$ la fonction finale, telle qu'elle se retrouve dans notre algorithme

aussi, afin de simplifier la compréhension, nous assignons parfois un scalaire à un vecteur, matrice ou tenseur. dans ce cas nous entendons simplement assigner la valeur scalaire à tous les éléments de la matrice, vecteur ou tenseur. nous appliquons également des fonction scalaire à des vecteurs, matrices ou tenseurs. cela signifie que la fonction est appliquée individuellement à chacun des éléments du tenseur. de cette façon pour une matrice x de dimension $n \times m$ nous avons

$$x = 0 \text{ est équivalent à } x = \begin{pmatrix} 0_{11} & \dots & 0_{1m} \\ \vdots & \ddots & \vdots \\ 0_{n1} & \dots & 0_{nm} \end{pmatrix}$$

$$y = f(x) \text{ nous donne } y = \begin{pmatrix} f(x_{11}) & \dots & f(x_{1m}) \\ \vdots & \ddots & \vdots \\ f(x_{n1}) & \dots & f(x_{nm}) \end{pmatrix}$$

5.1.6 Architecture

Afin d'être en mesure de bien évaluer les performances du modèle proposé, nous avons opté pour une architecture de base, soit une couche cachée connectée à une couche d'entrée et à une couche de sortie. La couche cachée est récurrente, ce qui veut dire qu'elle est pleinement connectée avec elle-même. Formellement, notre architecture A peut être définie avec $A = (X, W)$ où $X = \{X_0, X_1, X_2\}$, $W = \{W^{0,1}, W^{1,1}, W^{1,2}\}$. Nous avons choisi de retirer les connexions $W^{2,1}$ puisque celles-ci avaient un impact négatif sur la précision de notre modèle. Un réseau avec une seule couche cachée X^1 est désirable pour valider si notre approche présente du potentiel, car cela nous permet de vérifier que le réseau est en mesure d'apprendre une représentation des données exploitables sans les signaux dynamiques des couches supérieures qui pourrait venir brouiller les performances.

5.2 Proposition

La technique que nous avons développée est composée de trois couches. La première est la couche d'entrée, la seconde est la couche cachée qui vise à extraire une représentation des données et la dernière s'attarde à classifier cette représentation. Cette section présente en détails chacun des termes contenus dans la formalisation mathématique de celle-ci.

5.2.1 Activation

La fonction d'activation retenue est la fonction binaire. En plus de sa simplicité et de son origine biologique, nous l'avons également pour la facilité avec laquelle elle nous permet de faire des calculs impliquant seulement

ou excluant complètement les neurones actifs à un temps t . Notre fonction d'activation est $\phi(x)$ où x est le potentiel de membrane (l'état) d'un neurone ϕ est la fonction définie par:

$$\phi(x) = \begin{cases} 1 & \text{si } x > m_{\text{seuil}} \\ 0 & \text{si } x < m_{\text{seuil}} \end{cases} \quad (5.4)$$

Où m_{seuil} est le potentiel seuil de la membrane (au-delà duquel le neurone s'active). Nous dénotons $\Phi(X)$ l'application de ϕ à chacun des éléments d'un tenseur.

5.2.1.1 Durée cyclique cible

Puisque nous nous intéressons pour l'instant à la forme la plus simple de notre modèle, nous avons choisi le d'avoir pour cible un cycle minimal de période $N = 2$. Nous nous sommes arrêtés sur cette périodicité afin de valider notre approche pour plusieurs raisons. (1) Faciliter l'implémentation. Pour déterminer la cible d'apprentissage (section 5.2.1.4), avec $N = 2$ nous devons prendre en compte 3 itérations, soit $t - 1$, t et $t + 1$. En augmentant N nous devons également augmenter le nombre d'états à inclure dans le calcul de la cible. Il ne nous est pas évident à ce stade de quelle façon cela doit être fait et allonger le cycle représentait par conséquent un risque qui ne nous semblait pas nécessaire à ce stade de développement. (2) Augmenter la probabilité que notre méthode converge. Plus le cycle est long, plus la probabilité que des erreurs s'accumulent durant le cycle et que l'attracteur ne change de bassin d'attraction durant son exécution et que l'algorithme ne converge pas. (3) Faciliter l'analyse subséquente des dynamiques et des attracteurs présents dans le modèle entraîné.

5.2.1.2 Fixer l'entrée du réseau

Afin de générer une prédiction sur la classe associée à une image, nous devons dans un premier temps présenter cette image au réseau. L'image est d'abord aplatie de façon à prendre la matrice 28×28 pour en faire le vecteur `img` de taille 1×784 . Nous passons ensuite le vecteur `img` dans la fonction d'activation 5.4 avec $m_{\text{seuil}} = 0$ afin d'en faire un vecteur binaire. Par la suite, celui-ci est présenté au réseau avec

$$X^0 = \Phi(\text{img}) \quad (5.5)$$

Afin de faciliter l'obtention d'une dynamique cyclique dans le réseau nous avons opté d'alterner l'entrée entre les données du jeu et un vecteur nul. Puisque notre période cible est de $N = 2$, nous changeons l'entrée à chaque itération entre l'équation 5.7 et

$$X^0 = 0 \quad (5.6)$$

En d'autres mots, nous avons un compteur n qui est incrémenté à chaque itération. L'entrée clignote en changeant d'état à chaque itération (p. ex lorsque nous sommes à une itération n qui est paire, l'entrée est déterminée par la formule 5.7 et lorsque n est impaire l'entrée est le vecteur nul tel que décrit par l'équation 5.6. Formellement:

$$\iota(img_i) = \begin{cases} \phi(img_i) & \text{si l'itération courante (} n \text{) est paire} \\ 0 & \text{si l'itération courante (} n \text{) est impaire} \end{cases} \quad (5.7)$$

Nous dénotons $I(img)$ lorsque cette fonction est appliquée à chaque élément d'un vecteur.

5.2.1.3 Réfraction

Un attracteur fixe n'est pas l'objectif de notre entreprise et pourrait venir brouiller les résultats rendant difficile de cerner si les précisions obtenues sont attribuables aux dynamiques de notre algorithme ou simplement à de l'association Hebbienne classique. Afin d'éviter que le réseau ne tende vers celui-ci, nous avons implémenté Il s'agit également d'un élément qui ajoute à la similarité avec l'analogie biologique, tel que présenté à la section 3.4.1.

$$\rho(x_i^l[t]) = x_i^l[t] \circ (1 - \phi(x_i^l[t-1])) + m_{ref} * \phi(x_i^l[t-1]) \quad (5.8)$$

Ce qui nous donne en notation matricielle:

$$P(X[t]) = X[t] \circ (1 - \Phi(X[t-1])) + m_{ref} * \Phi(X[t-1]) \quad (5.9)$$

5.2.1.4 Cible d'apprentissage

L'algorithme que nous proposons ici est une forme de mémoire associative bi-directionnelle telle que présentée au chapitre 4. Par conséquent la règle d'apprentissage est basée sur la formule 4.7, dans laquelle on a le vecteur source X et le vecteur cible Y . Nous devons donc définir le vecteur cible Y pour la couche cachée ainsi que pour la couche de sortie, de façon à être en mesure de mettre à jour $W^{0,1}$, $W^{1,1}$ et $W^{1,2}$. Nous commençons par itérer le réseau pendant un certain temps pour le laisser converger, en utilisant l'équation 5.5 pour déterminer notre entrée.

Nous avons pris comme point de départ une règle de base qui détermine qu'elle est l'activation optimale de la couche cachée $X^1[t]$ pour obtenir l'activation observée sur $X^1[t+1]$. En d'autres termes, nous mettons à jour le réseau pendant n itérations pour lui laisser le temps de converger, puis nous tentons de renforcer l'attracteur dans lequel il a abouti:

$$\Theta'(X^l[t]) = W^{[l,k]} \times \Phi(X^k[t+1])^\top + W^{[l,l]} \times \Phi(X^l[t+1])^\top \quad (5.10)$$

Les neurones en état de réfraction, donc pour lesquels il est impossible qu'ils se soient activés au temps t peu importe leurs entrées, doivent être retirés de la cible, car cela reviendrait à d'optimiser la potentiation d'un neurone i au temps t pour que $\phi(x_i^l[t]) = 1$, alors que le neurone était activé au temps $t-1$, ce qui n'est pas autorisé par le principe de réfraction. Nous ajoutons donc un produit d'Hadamard avec $1 - \phi(x_i^l[t-1])$ donc un terme afin que $\theta'(x_i^l[t]) = 0$ si i est en état de réfraction au temps t de façon à ce qu'il n'y ait pas de modification pour ce neurone à cette itération:

$$\begin{aligned} \Theta''(X^l[t]) = & (W^{[l,k]} \times \Phi(X^k[t+1])^\top) + (W^{[l,l]} \times \Phi(X^l[t+1])^\top) \\ & \circ (1 - \Phi(X^l[t-1])) \end{aligned} \quad (5.11)$$

avec $k = l + 1$ dans notre cas.

Le premier terme concerne le signal venant de la couche supérieure, alors que le second est celui provenant

dans la couche courante, via les connexions récurrentes. Finalement, nous ajoutons un terme pour l'apprentissage anti-Hebbien afin de réduire les connexions entre des neurones activés au temps t vers ceux qui l'étaient au temps $t - 1$:

$$\begin{aligned} \Theta(X^l[t]) = & ((W^{[l,k]} \times \Phi(X^k[t+1])^\top) + (W^{[l,l]} \times \Phi(X^l[t+1])^\top)) \\ & \circ (1 - \Phi(X^l[t-1])) - (m_{ref} * \Phi(X^l[t-1])) \end{aligned} \quad (5.12)$$

5.2.1.5 Énergie

Dans le but d'implémenter une compétition implicite entre les neurones, nous avons introduit le concept de l'énergie qui vise à moduler les changements d'état en fonction de la fréquence d'activation du neurone par le passé. Pour ce faire au moment de calculer le δ d'activation d'une itération à la suivante, nous multiplions le δ par une valeur, que nous appelons l'énergie de ce neurone. Formellement, pour un neurone x_i au temps t , son énergie e_{x_i} au temps $t + 1$ est donnée par $e_{x[t+1]} = e_{x[t]} + \epsilon(x[t])$ avec la fonction ϵ définie par:

$$\delta\epsilon(x) = \begin{cases} e^- & \text{si } x > m_{seuil} \\ e^+ & \text{si } x < m_{seuil} \end{cases} \quad (5.13)$$

En d'autres termes, l'énergie du neurone est réduite de $|e^-|$ lorsqu'il est activé et récupère e^+ lorsqu'il ne l'est pas. La valeur initiale de l'énergie est de 1.0.

5.2.1.6 Inférence

L'équation de base pour la mise à jour des états consécutifs est la même que pour l'apprentissage dans les mémoires associatives à l'exception que le second terme est donné par la cible d'apprentissage au lieu d'être le vecteur d'état du réseau:

$$X^l[t+1]' = X^l[t] + \Phi(X^{[l-1]}[t]) \times W^{l-1,l} + \Phi(X^l[t]) \times W^{l,l} \quad (5.14)$$

Nous y ajoutons ensuite la réfraction qui met des neurones activés à l'itération précédente à m_{ref} :

$$X^l[t+1]'' = ((X^l[t] + \Phi(X^{l-1}[t]) \times W^{l-1,l} + \Phi(X^l[t]) \times W^{l,l}) \circ (1 - \Phi(X^l[t-1]))) + (m_{ref} * \Phi(X^l[t-1])) \quad (5.15)$$

Finalement, nous avons le terme qui module l'activation en fonction de l'énergie de chaque neurone pour obtenir:

$$X^l[t+1] = (X^l[t] + (\Phi(X^{l-1}[t]) \times W^{l-1,l} + \Phi(X^l[t]) \times W^{l,l}) \circ \epsilon(X^l[t])) \circ (1 - \Phi(X^l[t-1])) + (m_{ref} * \Phi(X^l[t-1])) \quad (5.16)$$

5.2.1.7 Apprentissage

La version de base pour l'apprentissage est donc:

$$\Delta W^{l,k}[t] = \lambda * \Phi(X^l[t-1]) \times \Theta(X^k[t])^\top \quad (5.17)$$

Pour forcer 5.17 l'apprentissage d'un cycle de période $N = 2$ au sein des couches récurrentes, lorsque $l = k$:

$$\Delta W^{l,l}[t] = \lambda * (\Phi(X^l[t-1]) \times \Theta(X^l[t])^\top + \Theta(X^l[t-1]) \times \Phi(X^l[t])^\top) \quad (5.18)$$

et l'équation 5.17 est inchangée lorsque $l \neq k$.

5.2.1.8 Détermination initiale du signe des poids

Les poids sont initialisés en suivant une distribution gaussienne centrée sur 0 afin de balancer les forces excitantes et inhibantes dans le réseau. Une fois cette initialisation faite, le signe des connexions (positif ou négatif) est fixé pour la suite afin de maintenir cet équilibre. Pour ce faire, nous ajoutons un terme pour multiplier par $\tanh(W)$. Cela a également pour effet d'éviter qu'une valeur δw ne soit trop grande, augmentant la stabilité de l'apprentissage:

$$W'[t + 1] = \tanh(W[t - 1]) \circ (W + \Delta W[t]) \quad (5.19)$$

5.2.1.9 Bloquer la valeur des poids

Finalement, afin d'éviter que les valeurs des poids n'explose avec le temps, ce qui entraverait l'apprentissage, nous appliquons une fonction $\text{clip}(w_{i,j}^{[l,k]})$ à tous les poids

Ce qui nous donne:

$$W[t] = \text{clip}(W'[t]) \quad (5.20)$$

avec la fonction $\text{clip}(x)$ définie comme:

$$\text{clip}(x) = \begin{cases} x & \text{si } x \geq p_{\min} \text{ et } x \leq p_{\max} \\ p_{\min} & \text{si } x < p_{\min} \\ p_{\max} & \text{si } x > p_{\max} \end{cases} \quad (5.21)$$

On peut donc voir que l'on apprend ici un cycle allant de l'activation précédente vers la cible et de la cible vers l'activation précédente. Lors de première implémentation, le second terme de l'addition était absent, car nous voulions favoriser une émergence naturelle de cycles limites. Cependant, après quelques expériences comparatives, nous avons constaté que l'ajout de ce terme améliorait significativement les performances du modèle (+ 10%) à l'évaluation. Nous avons donc opté de l'inclure dans la version présentée ici, mais

nous y reviendrons au moment de la discussion afin de proposer nos hypothèses concernant les facteurs derrière cette disparité ainsi que nos proposition pour le retirer dans le futur afin de permettre des cycles plus long que $n = 2$.

Tableau 5.1 Paramètres

Nom du paramètre	Description	Valeur
m_{seuil}	Le seuil au-delà duquel un neurone s'active	0.0
m_{min}	La valeur minimum de l'état d'un neurone	-0.8
m_{max}	La valeur maximale de l'état d'un neurone	0.5
p_{min}	La valeur minimale qu'un poid peut prendre	-0.1
p_{max}	La valeur maximale qu'un poid peut prendre	0.1
λ	Taux d'apprentissage	0.035
bruit	Le bruit stochastique pour l'apprentissage	0.015
e^+	δe lorsque le neurone n'est pas activé	0.04
e^-	δe lorsque le neurone est activé	-0.2

5.2.2 Paramètres globaux

Les paramètres globaux utilisés pour obtenir les résultats exposés dans la figure 5.2 sont présentés dans le tableau 5.1. Une fois que nous avons eu une implémentation fonctionnelle de notre algorithme, nous avons effectué de nombreuses séries d'entraînements automatisées variant les différents paramètres définis plus haut afin de trouver la valeur optimale pour chacun d'entre eux.

5.3 Résultats

Suite à cette recherche, nous avons obtenu les paramètres présentés dans le tableau 5.1. Le graphique 5.2 relate l'évolution de la précision sur le jeu de donnée d'évaluation durant le processus d'entraînement. Celle-ci cessait de s'améliorer entre 30 000 et 50 000 images d'entraînement, et la meilleure performance obtenue à l'issue de l'apprentissage était un taux d'erreur de 8.6%. On peut y voir que notre proposition de bio-RNN est en mesure d'apprendre avec un nombre limité d'exemples, mais semble plafonner lorsque le taux d'erreur se situe entre 12% et 8%.

Tout d'abord, le tracé en mauve représente l'évolution de la performance d'une simple mémoire associative bi-directionnelle à 1024 neurones. Celle-ci fait office de point de référence, puisqu'il s'agit de l'algorithme standard qui se rapproche le plus de notre proposition. Par conséquent, il s'agissait pour nous de la performance de base que nous souhaitions atteindre initialement. On peut voir que la performance plafonne rapidement à 67% avant de se détériorer progressivement. Nous sommes parvenus à atteindre cet objectif assez rapidement lors du développement de notre algorithme, et nous nous sommes par la suite tournés vers la RP comme étalon d'évaluation.

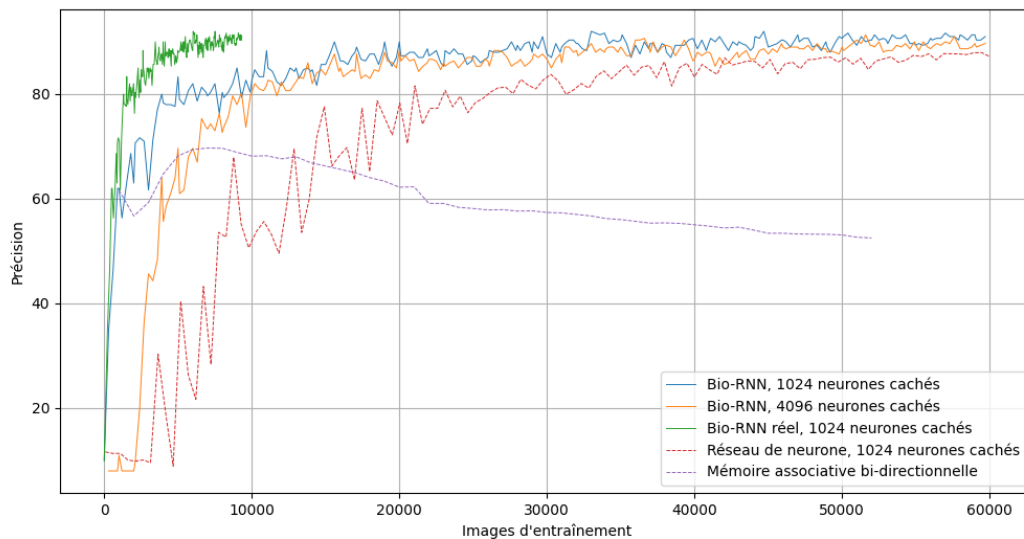


Figure 5.2 Évolution de l'apprentissage de notre algorithme bio-RNN

Nous avons testé trois versions différentes de notre algorithme. Nous avons sélectionné le réseau à 1024 neurones dans la couche cachée avec la RP provenant de nos expériences présentées dans la figure 5.1 comme point de comparaison principal. Cette décision est motivée par le fait qu'il s'agit de l'architecture la plus près de celle que nous avons utilisée pour une grande majorité de nos tests avec notre bio-RNN, soit celle avec 1024 neurones dans la couche cachée. Les performances de cette dernière sont illustrées à l'aide du tracé en bleu dans la figure 5.2. On peut voir que notre approche apprend à un rythme plus élevé que la RP et maintient son avance sur toute la première époque (passage complet du jeu de donnée d'entraînement). À terme, la RP continue de s'améliorer alors que notre approche plafonne et bien que nous comptons tenter d'améliorer ce résultat dans des recherches futures, il n'en demeure pas moins intéressant dans notre contexte puisque la rapidité de l'apprentissage était l'une des métriques principales qui nous intéressaient.

Le tracé en vert est le résultat d'une légère modification de notre algorithme tel que présenté dans les sections précédentes. Avant d'effectuer l'apprentissage sur une entrée donnée, nous vérifions comment le réseau la classifie. Si l'inférence s'avère prédire une classe erronée, nous exécutons par la suite normalement l'algorithme de mise à jour des poids. En contrepartie, si la classification est juste, nous passons directement au prochain exemple sans modifier les poids.

C'est la raison pour laquelle les données s'arrêtent juste avant les 10 000 images d'entraînement. Simplement l'époque s'est terminée avec le modèle ayant traité seulement 9317 images sur les 60 000 du jeu de données avec un taux d'erreur final de 9.2%. Cette mesure ne s'est pas avérée avoir d'impact sur la performance, mais elle a permis de démontrer qu'avec un nombre limité de données de qualité, couvrant la distribution du jeu de données et étant suffisamment orthogonales les unes par rapport aux autres, notre algorithme est en mesure d'apprendre drastiquement plus rapidement. Nous obtenons une précision qui progresse deux ordres de magnitudes plus vite que la RP. Cela démontre un potentiel énorme pour notre algorithme s'il est en mesure d'affiner sa performance globale.

L'absence de modèles bio-RNN avec moins de neurones est délibérée. Lors de nos expériences, la performance diminuait linéairement en fonction de la réduction de la taille de la couche cachée en deçà de 1024. Ce résultat n'était pas surprenant outre mesure, puisqu'en raison des dynamiques cycliques que nous tentons d'induire, nous nous attendions à ce qu'un nombre minimal de noeuds soit nécessaire pour que des cycles variés émergent naturellement.

Un autre point à relever est qu'avec l'accroissement de la taille du réseau, on voit une certaine diminution de la vitesse d'apprentissage, mais celle-ci est moins notable que pour la RP, tel qu'illustré dans le graphique 5.1. Également, les performances sont beaucoup plus stables avec une variation nettement inférieure lors de la progression de l'entraînement. Ces deux éléments sont fort intéressants car ils supposent une meilleure mise en échelle des bio-RNN que de la RP. Avec les modèles massifs qui sont désormais la norme, souvent dans les milliards de paramètres et dépassant parfois le billion, des améliorations même marginales à ces niveaux peuvent entraîner des économies significatives en termes de ressources pour l'entraînement des modèles de pointe. Cela permettrait de développer des modèles plus large et plus performant à coût égal ou des modèles similaires à moindre coût.

5.3.1 Analyse visuelle

Une analyse visuelle des poids $W^{0,1}$ suite à l'apprentissage, que l'on peut retrouver dans la figure 5.3, révèle plusieurs éléments intéressants. Chaque image s'y retrouvant représente un ensemble de connexions allant de l'entrée à un neurone de la couche caché, soit appartenant à $W^{0,1}$. En d'autres termes il s'agit de champs récepteurs de neurones cachés sur l'entrée. À noter qu'il s'agit d'un sous-ensemble des neurones de la couche cachée illustré à titre indicatif, leur grand nombre ne permettant pas de les afficher entièrement de façon lisible.

Tout d'abord, on y reconnaît qu'un apprentissage sensé prend définitivement place et que chaque neurone apprend à s'activer suivant des caractéristiques abstraites de l'entrée, tel que prévu. Également, on peut y voir que dans les zones extérieures, qui ne contiennent généralement pas d'informations (les pixels qui sont presque toujours à 0 dans les jeux d'entraînement et d'évaluation), aucun apprentissage ne prend place.

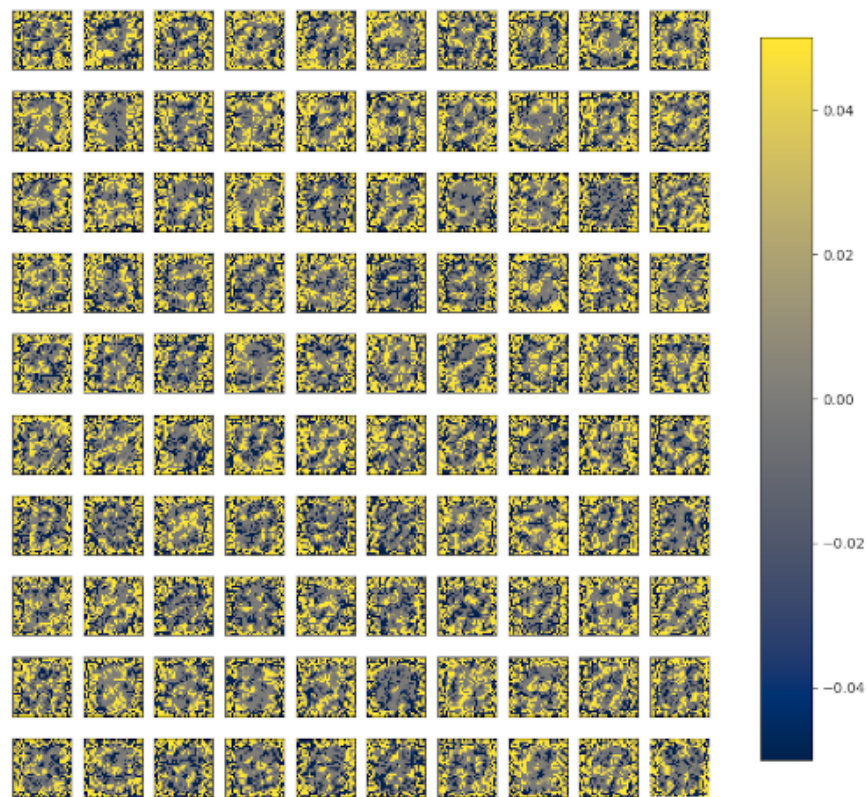


Figure 5.3 Les matrices de $W_{0,1}$ pour un ensemble de neurones après l'apprentissage

Nous avons tenté divers moyens pour pallier à ceci, puisqu'intuitivement ces poids devraient tendre vers 0 afin que lorsqu'une image sort légèrement de la distribution des données d'entraînement, elle ait plus de

probabilité d'être néanmoins bien classifiée (p. ex. si le nombre dans une image est légèrement décalé vers la gauche ou la droite, des pixels actifs de cette image vont activer des neurones qui ne devraient pas l'être en raison de ces connexions qui sont demeurées inchangées depuis l'initialisation).

On peut observer que les images générées par les poids connectant les neurones à la couche d'entrée semblent être bruitées dans leur périphérie. Il semblerait que ces poids, qui sont sous-utilisés puisque très peu activés par les données d'entrées, ne sont pas modifiés par le réseau lors de l'entraînement puisque qu'aucun signal n'y transige. Ils restent donc hautement similaires à leur configuration initiale, ce qui entraîne ce phénomène. Cette observation est encourageante dans le cadre de notre recherche, car cela semble suggérer d'une part que l'oubli catastrophique de la RP pourrait être mitigé par notre approche puisque les apprentissages anciens demeurent cristallisés dès lors que la distribution des données d'entrées est suffisamment différente de celles apprises au préalable. Dans un second temps, cela suggère également que le réseau dispose de connexions inutilisées pouvant être mises à profit dans l'apprentissage de données qui diffèrent de la distribution initiale. Plus de recherches seront nécessaires pour valider ces postulats.

Les images de la figure 5.3.1 illustrent quand à elles un sous-ensemble du champs récepteur des neurones cachés sur les autres neurones cachées. Il s'agit donc d'une visualisation des connexions récurrentes de $W^{1,1}$. On y observe, dans un premier temps, qu'il n'y a que très peu de poids positifs (en jaune). Cela découle de 5.7. Lorsque l'entrée est nulle, il est normal que peu de neurones soient activés dans la couche cachée, ceux qui le sont n'étant qu'activer par les connexions récurrente. Nous croyons également que cela est dû au fait que lorsque l'entrée est présente (équation 5.5) cela stimule un grand nombre de neurones à s'activer. Par conséquent, substantiellement moins de neurones sont réceptif à être activer en raison de la réfraction (5.9).

5.4 Optimisations

L'algorithme présenté ici est le résultat final d'un processus d'optimisation qui s'est étendu sur de nombreuses itérations. Au fil de ce développement, de nombreuses variations de l'algorithme initial ont été méthodiquement testées dans le but d'en affiner la performance et l'efficacité.

Plusieurs de ces variations incluait des éléments bio-plausibles que nous croyions initialement judicieux d'intégrer, mais que nous avons finalement retirés puisqu'ils impactaient négativement la précision lors de l'inférence. Cela n'infirme pas complètement leur pertinence pour des recherches futures mais, comme

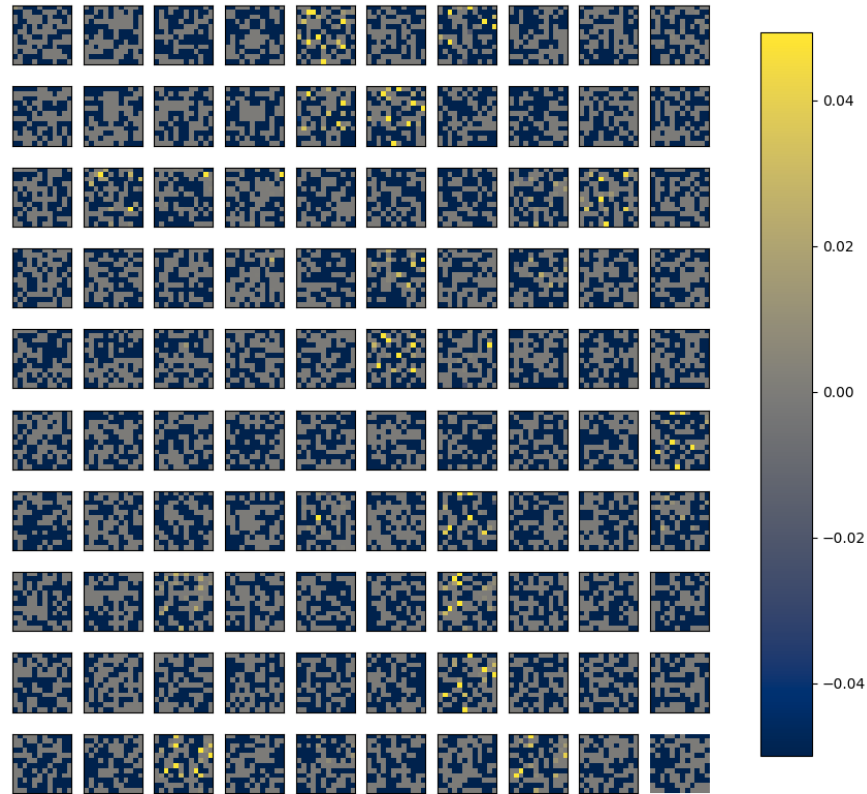


Figure 5.4 Les matrices de $W_{1,1}$ pour un ensemble de neurones après l'apprentissage

nous l'avons, l'apprentissage biologique est un processus excessivement complexe sur lequel nous en connaissons toujours bien peu. En pratique, cela signifie qu'un principe donné A peut n'être valable que lorsqu'un mécanisme complémentaire B est présent. Il se peut également que l'abstraction que nous en avons fait soit trop simpliste ou ne soit simplement pas appropriée dans le contexte. Un exemple est la taille des cycles abordé à la section 5.4.1. Il se peut qu'il existe un motif particulier de connectivité, ou des propriétés biochimiques de certains neurotransmetteurs, qui doivent être présent avec une concentration précise dans le liquide céphalo-rachidien, qui régule l'émergence, la période et la synchronisation des attracteurs temporels. Or, nous n'avons aucun moyen à ce stade d'être en mesure de déterminer ce genre de dépendance pour les intégrer dans notre modèle. Nous nous sommes donc résolus à simplement retirer ces éléments qui venaient substantiellement impacter négativement la performance de notre proposition. Il n'en demeure que ces éléments représentent des questions de recherche ouvertes sur lesquelles nous comptons revenir lors des recherches ultérieures.

D'autres se sont révélées particulièrement notables puisqu'elles démontrent la pertinence de certains choix

qui ont été faits dans l'implémentation finale. Finalement, certains des détails d'implémentations n'ont pas été couverts dans la formalisation mathématique présentée car nous avons jugé qu'il était plus judicieux de les exposer suite à la présentation des résultats. Par conséquent, cette section vise à fournir un aperçu de ces variantes significatives, mettant en lumière les ajustements et les améliorations qui ont été apportées au fur et à mesure de l'avancement de nos recherches.

5.4.1 Cycles forcés

Nous avons tenté de retirer l'apprentissage explicite des cycles défini dans l'équation 5.18 afin de laisser le modèle apprendre par lui-même les cycles qu'il désirait avec une période arbitraire, car nous n'avons croisé aucune base théorique qui dicte que nous devons explicitement imposer le cycle au moment de l'apprentissage. Nous avons même comme hypothèse initiale que pour une entrée donnée le réseau pouvait vagabonder entre différents attracteurs et qu'il pouvait y avoir une valeur à le faire, lui permettant de mieux discriminer les images améliorant ainsi la classification. Cependant, après avoir comparé expérimentalement les deux approches, il en est ressorti qu'il était optimal de forcer l'apprentissage d'un cycle oscillatoire de période $N = 2$, avec une amélioration de la précision d'environ 10% – 15% au moment de l'évaluation.

5.4.2 Détermination initial du signe des poids

La détermination initiale des poids est une contrainte que nous avons incluse dès le départ dans notre modèle. Nous avons inclus cette contrainte en raison de sa contrepartie biologique, où un neurone est soit de type "excitant" soit de type "inhibiteur". Cependant nous n'avons pas jugé utile de l'implémenter au niveau des neurones en soi et nous nous sommes contentés de les fixer au niveau des connexions individuelles, ne voyant pas de différence flagrante au niveau logique et mathématique qui justifierait de le faire. Durant nos recherches, le retrait de cette contrainte a fait l'objet d'expérimentations avec pour résultat un effondrement complet de l'apprentissage, ce qui est venu nous confirmer dans notre décision.

5.4.3 Affaiblissement temporel des poids

Un autre de ces éléments, que nous avons initialement jugé pertinent d'implémenter dans notre algorithme et qui est documenté comme processus biologique, est le déclin temporel des synapses inutilisées. Nous avons tenté de l'inclure sous diverses formes, entres autres afin de faire tendre les connexions provenant de la périphérie de l'image vers 0. Puisque celles-ci ne devraient pas avoir d'impact sur l'activation ou l'inhibition

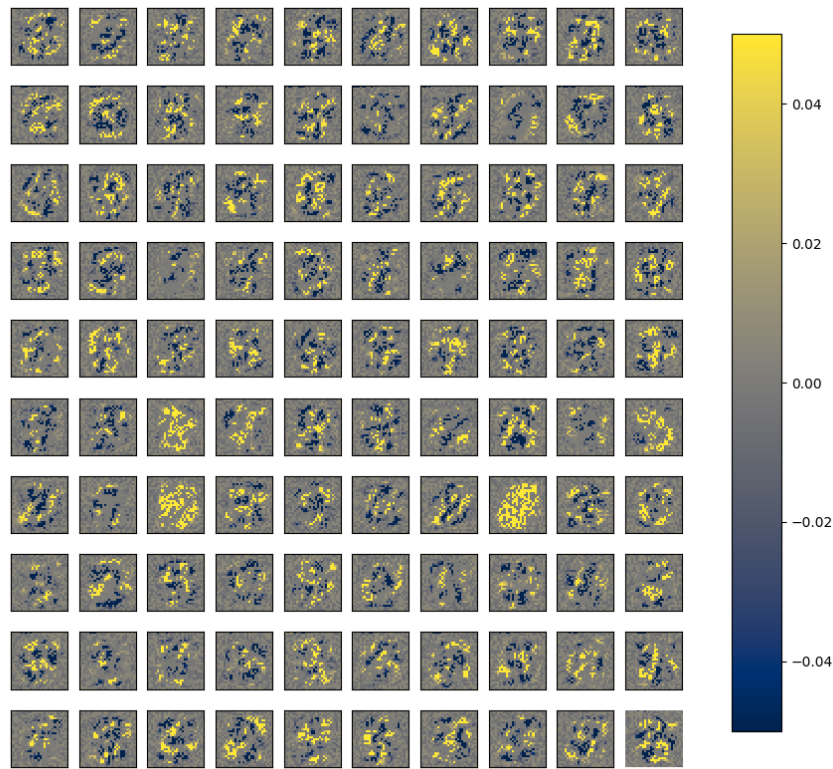


Figure 5.5 Les matrices de $W_{0,1}$ pour un ensemble de neurones après l'apprentissage, avec un affaiblissement temporel des poids

des neurones de la couche cachée, nous croyions que les affaiblir au fil du temps mènerait à une meilleure précision du modèle.

Cependant ces tentatives sont demeurées infructueuses. Au mieux ayant un impact légèrement négatif sur la précision du modèle et dans les cas les plus désastreux menant à un effondrement complet des performances après environs 5000–15000 itérations d'apprentissage. La figure 5.5 est tiré d'une expérience implémentant cette mesure et illustre l'effet de cette mesure sur les poids d'entrée du modèle. Visuellement, cela semble faire plus de sens que les poids tirés de nos résultats illustrés dans la figure 5.3 mais en pratique la performance atteint un plafond inférieur (et plus rapidement), soit 79%, contre 90% sans cette mesure. Cependant en raison de sa provenance biologique, nous croyons qu'une optimisation de la sorte représente une avenue expérimentale intéressante qui peut justifier des recherches futures dans cette direction.

CHAPITRE 6
CONCLUSION

Il est indéniable que la rétropropagation a été un moteur essentiel dans les progrès de l'IA, permettant la résolution de problèmes autrefois intraitables et l'atteinte de performances de pointe dans de nombreuses tâches. Cependant, nous croyons, en raison des arguments qui ont été présentés, qu'il est désirable d'explorer des alternatives plus proches des mécanismes d'apprentissage biologiques. L'apprentissage Hebbien et anti-Hebbien, la STDP, les systèmes dynamiques et les mémoires associatives, en tant que représentants de ces alternatives, offrent des perspectives intéressantes pour le développement de nouveaux algorithmes. Inspirés par les principes biologiques de la plasticité synaptique, ils nous éclairent sur des approches plus naturelles et potentiellement plus efficaces pour effectuer l'apprentissage dans des réseaux de neurones récurrents, bien qu'ils aient été historiquement relégués au second plan par rapport à la rétropropagation en raison de leurs performances inférieures dans de nombreux contextes, ce qui nous a poussés à sélectionner cette approche comme point de départ pour la création de notre algorithme.

Dans ce mémoire nous avons vu comment le cerveau humain peut être décrit comme un système dynamique qui alterne constamment entre différents états sans nécessairement atteindre un équilibre stable. Cette flexibilité, due à sa nature chaotique intrinsèque, lui permet de réagir de manière adaptative aux stimuli internes et externes. Cette caractéristique est cruciale pour les fonctions cognitives, où le cerveau doit être en mesure de passer rapidement d'un état à un autre pour encoder et récupérer des informations. Nous avons également présenté divers algorithmes pertinents dans notre contexte, sur lesquels nous nous sommes basés pour concevoir notre approche.

À travers une approche à la fois théorique et expérimentale, nous avons exploré les fondements de différents paradigmes d'apprentissage biologiques et artificiels, ainsi que leur application potentielle dans des architectures neuronales récurrentes dynamiques. Notre proposition d'un algorithme d'apprentissage bio-inspiré pour les RNN représente un pas dans cette direction en démontrant la faisabilité et l'efficacité possible d'une fusion entre ces différentes approches. Pour ce faire, nous avons formalisé un nouvel algorithme d'apprentissage inspiré par la biologie qui a été présenté en détails. Inspiré du fonctionnement général des mémoires associatives et basé sur les travaux de (Molter *et al.*, 2005), nous avons généralisé ces approches à plusieurs couches et avons implémenté une version simple à 3 couches afin de démontrer expérimentalement sa validité.

Bien que notre modèle offre des performances globales limitées, ne parvenant pas à surclasser un grand nombre de techniques ML sur la classification d'image, nous sommes néanmoins parvenus à atteindre

certains des objectifs que nous nous étions fixés.

Dans un premier temps, notre modèle démontre des capacités d'apprentissage rapides comme nous l'espérions, étant en mesure d'atteindre une performance de 80.5% après avoir été entraîné sur seulement 1505 exemples, contre plus de 21000 pour notre implémentation comparative d'ANN avec propagation arrière. Cependant, malgré cette supériorité initiale de notre modèle, notre approche demeure inférieure en termes absolus. En effet, alors que notre proposition sature à autour de 40 000 itérations, la rétropropagation continue de s'améliorer sur plusieurs époques, pour éventuellement atteindre un taux d'erreur $< 4\%$. Cependant la RP ne parvenant pas à surpasser notre algorithme en 1 époque. Cependant, si nous entraînons les modèles pendant plusieurs époques, éventuellement les modèle avec la RP finissent toujours par reprendre le dessus, obtenant éventuellement un taux d'erreur entre 5% et $< 1\%$ tout dépendant de l'architecture évaluée et du nombre d'époques d'entraînement.

Dans un second temps, nous avons atteint notre second objectif, parvenu à encoder le traitement de l'information à l'intérieur de cycles limites. Ainsi faisant, nous avons pu démontrer que ce type de dynamique dispose des capacités nécessaires pour soutenir l'apprentissage dans des réseaux de neurones récurrents et qu'il s'agit par conséquent d'une piste de recherche viable.

Finalement, nos différentes expériences nous ont également permis d'évaluer différents concepts biologiques dans le cadre de leur application aux RNN. Cela nous a permis d'apprendre que de fixer les poids comme étant inhibiteur ou excitant à l'initialisation menait à un apprentissage plus robuste menant à une diminution du taux d'erreur pour la tâche de classification. Nous avons également pu constater que l'affaiblissement temporel des poids était contre productif dans ce type de contexte expérimental.

Bien que nos résultats préliminaires si situent en-deçà de l'état de l'art, nous sommes néanmoins convaincus qu'il s'agit d'une approche prometteuse qui mérite d'être poussée plus loin. Au fil de nos recherches, nous avons implémenté différentes variantes de notre algorithme afin de tester l'impact que pouvaient avoir différents paramètres et différentes formalisations sur la performance. De nombreuses autres ont été considérées, mais leurs mises en pratique dépassaient le cadre du présent travail. Elles représentent néanmoins des pistes d'amélioration envisageables pour nos recherches futures. L'utilisation de cycles plus longs, l'implémentation de règles favorisant la synchronisation de groupes de neurones, l'implémentation de la régression logistique pour l'optimisation des connexions de la dernière couche cachée à la couche

de sortie ainsi que l'ajout de couches récurrentes latérales, donc étant connecté bi-directionnellement avec seulement une autre couche sont toutes des pistes que nous considérons explorer dans nos travaux futurs.

BIBLIOGRAPHIE

- Anand, K. S. et Dhikav, V. (2012). Hippocampus in health and disease: An overview. *Ann. Indian Acad. Neurol.*, 15(4), 239–246.
- Anderson, J. R. (2007). *How Can the Human Mind Occur in the Physical Universe?* Oxford University Press.
<http://dx.doi.org/10.1093/acprof:oso/9780195324259.001.0001>. Récupéré de
<https://doi.org/10.1093/acprof:oso/9780195324259.001.0001>
- Anisimova, M., van Bommel, B., Wang, R., Mikhaylova, M., Wiegert, J. S., Oertner, T. G. et Gee, C. E. (2022). Spike-timing-dependent plasticity rewards synchrony rather than causality. *Cereb. Cortex*, 33(1), 23–34.
- Asghar, M. S., Arslan, S. et Kim, H. (2021). A low-power spiking neural network chip based on a compact lif neuron and binary exponential charge injector synapse circuits. *Sensors*, 21(13).
<http://dx.doi.org/10.3390/s21134462>. Récupéré de
<https://www.mdpi.com/1424-8220/21/13/4462>
- Barreto, E. et Cressman, J. R. (2011). Ion concentration dynamics as a mechanism for neuronal bursting. *J. Biol. Phys.*, 37(3), 361–373.
- Bennett, M. R., Gibson, W. G. et Robinson, J. (1994). Dynamics of the CA3 pyramidal neuron autoassociative memory network in the hippocampus. *Philos. Trans. R. Soc. Lond. B Biol. Sci.*, 343(1304), 167–187.
- Berner, R., Gross, T., Kuehn, C., Kurths, J. et Yanchuk, S. (2023). Adaptive dynamical networks. *Physics Reports*, 1031, 1–59. Adaptive dynamical networks,
<http://dx.doi.org/https://doi.org/10.1016/j.physrep.2023.08.001>. Récupéré de
<https://www.sciencedirect.com/science/article/pii/S0370157323002685>
- Bilder, R. M. et Knudsen, K. S. (2014). Creative cognition and systems biology on the edge of chaos. *Frontiers in Psychology*, 5. <http://dx.doi.org/10.3389/fpsyg.2014.01104>. Récupéré de
<https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2014.01104>
- Bob, P. (2007). Chaos, brain and divided consciousness. *Acta Univ. Carol. Med. Monogr.*, 153, 9–80.
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I. et Amodei, D. (2020). Language models are few-shot learners. Dans H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, et H. Lin (dir.). *Advances in Neural Information Processing Systems*, volume 33, 1877–1901. Curran Associates, Inc. Récupéré de
https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfc4967418bfb8ac142f64a-Paper.pdf
- Buonomano, D. et Carvalho, T. (2009). Spike-timing-dependent plasticity (stdp). In L. R. Squire (dir.), *Encyclopedia of Neuroscience* 265–268. Oxford: Academic Press

- Bush, D., Philippides, A., Husbands, P. et O'Shea, M. (2010). Reconciling the STDP and BCM models of synaptic plasticity in a spiking recurrent neural network. *Neural Comput.*, 22(8), 2059–2085.
- Byerly, A., Kalganova, T. et Dear, I. (2020). A branching and merging convolutional network with homogeneous filter capsules.
- Censor, N., Dimyan, M. A. et Cohen, L. G. (2010). Modification of existing human motor memories is enabled by primary cortical processing during memory reactivation. *Curr. Biol.*, 20(17), 1545–1549.
- Czaczkes, T. J. et Heinze, J. (2015). Ants adjust their pheromone deposition to a changing environment and their probability of making errors. *Proceedings of the Royal Society B: Biological Sciences*, 282(1810), 20150679. <http://dx.doi.org/10.1098/rspb.2015.0679>. Récupéré de <https://royalsocietypublishing.org/doi/abs/10.1098/rspb.2015.0679>
- Danforth, C. (2013). Chaos in an atmosphere hanging on a wall – mpe.dimacs.rutgers.edu. <http://mpe.dimacs.rutgers.edu/2013/03/17/chaos-in-an-atmosphere-hanging-on-a-wall/>. [Accessed 13-03-2024].
- DAVID MORGAN, E. (2009). Trail pheromones of ants. *Physiological Entomology*, 34(1), 1–17. <http://dx.doi.org/https://doi.org/10.1111/j.1365-3032.2008.00658.x>. Récupéré de <https://resjournals.onlinelibrary.wiley.com/doi/abs/10.1111/j.1365-3032.2008.00658.x>
- Decot, H. K., Namboodiri, V. M. K., Gao, W., McHenry, J. A., Jennings, J. H., Lee, S.-H., Kantak, P. A., Jill Kao, Y.-C., Das, M., Witten, I. B., Deisseroth, K., Shih, Y.-Y. I. et Stuber, G. D. (2017). Coordination of Brain-Wide activity dynamics by dopaminergic neurons. *Neuropsychopharmacology*, 42(3), 615–627.
- Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6), 141–142.
- Devlin, J., Chang, M.-W., Lee, K. et Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. Dans *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186., Minneapolis, Minnesota. Association for Computational Linguistics. <http://dx.doi.org/10.18653/v1/N19-1423>. Récupéré de <https://www.aclweb.org/anthology/N19-1423>
- Faisal, A. A., Selen, L. P. J. et Wolpert, D. M. (2008). Noise in the nervous system. *Nat. Rev. Neurosci.*, 9(4), 292–303.
- Feldman, D. E. (2012). The spike-timing dependence of plasticity. *Neuron*, 75(4), 556–571.
- Feng, J. et Tirozzi, B. (1997). Capacity of the hopfield model. *Journal of Physics A: Mathematical and General*, 30(10), 3383. <http://dx.doi.org/10.1088/0305-4470/30/10/016>. Récupéré de <https://dx.doi.org/10.1088/0305-4470/30/10/016>
- Fink, A. et Benedek, M. (2014). EEG alpha power and creative ideation. *neurosci. biobehav. rev.*, 44, 111–123.

- Fischer, I., Vicente, R., Buldou, J. M., Peil, M., Mirasso, C. R., Torrent, M. C. et Garcia-Ojalvo, J. (2006). Zero-lag long-range synchronization via dynamical relaying. *Phys. Rev. Lett.*, 97, 123902. <http://dx.doi.org/10.1103/PhysRevLett.97.123902>. Récupéré de <https://link.aps.org/doi/10.1103/PhysRevLett.97.123902>
- Folli, V., Leonetti, M. et Ruocco, G. (2017). On the maximum storage capacity of the hopfield model. *Frontiers in Computational Neuroscience*, 10. <http://dx.doi.org/10.3389/fncom.2016.00144>
- Gao, W.-J. et Goldman-Rakic, P. S. (2003). Selective modulation of excitatory and inhibitory microcircuits by dopamine. *Proceedings of the National Academy of Sciences*, 100(5), 2836–2841. <http://dx.doi.org/10.1073/pnas.262796399>. Récupéré de <https://www.pnas.org/doi/abs/10.1073/pnas.262796399>
- Gao, Z., Dang, W., Wang, X., Hong, X., Hou, L., Ma, K. et Perc, M. (2021). Complex networks and deep learning for EEG signal analysis. *Cogn. Neurodyn.*, 15(3), 369–388.
- Gelblum, A., Fonio, E., Rodeh, Y., Korman, A. et Feinerman, O. (2020). Ant collective cognition allows for efficient navigation through disordered environments. *Elife*, 9.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. et Bengio, Y. (2020). Generative adversarial networks. *Commun. ACM*, 63(11), 139–144.
- Gosti, G., Folli, V., Leonetti, M. et Ruocco, G. (2019). Beyond the maximum storage capacity limit in hopfield recurrent neural networks. *Entropy (Basel)*, 21(8), 726.
- Graupner, M. et Gutkin, B. (2009). Modeling nicotinic neuromodulation from global functional and network levels to nAChR based mechanisms. *Acta Pharmacologica Sinica*, 30(6), 681–693.
- Graves, A., rahman Mohamed, A. et Hinton, G. E. (2013). Speech recognition with deep recurrent neural networks. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 6645–6649. Récupéré de <https://api.semanticscholar.org/CorpusID:206741496>
- Gutzwiller, M. C. (1998). Moon-earth-sun: The oldest three-body problem. *Rev. Mod. Phys.*, 70, 589–639. <http://dx.doi.org/10.1103/RevModPhys.70.589>. Récupéré de <https://link.aps.org/doi/10.1103/RevModPhys.70.589>
- Ha, S.-Y., Noh, S. E. et Park, J. (2016). Synchronization of kuramoto oscillators with adaptive couplings. *SIAM Journal on Applied Dynamical Systems*, 15(1), 162–194. <http://dx.doi.org/10.1137/15M101484X>. Récupéré de <https://doi.org/10.1137/15M101484X>
- Hafting, T., Fyhn, M., Molden, S., Moser, M.-B. et Moser, E. I. (2005). Microstructure of a spatial map in the entorhinal cortex. *Nature*, 436(7052), 801–806.
- Hammond, C. (2015). Chapter 4 - the voltage-gated channels of na⁺ action potentials. In C. Hammond (dir.), *Cellular and Molecular Neurophysiology (Fourth Edition)* 55–91. Boston: Academic Press, (fourth edition éd.)

- He, K., Zhang, X., Ren, S. et Sun, J. (2016). Identity mappings in deep residual networks. Dans B. Leibe, J. Matas, N. Sebe, et M. Welling (dir.). *Computer Vision – ECCV 2016*, 630–645., Cham. Springer International Publishing.
- Hebb, D. O. (1949). *The organization of behavior*. New York, USA: Wiley.
- Hertz, J., Krough, A. et Palmer, G., R. (1991). *Introduction To The Theory Of Neural Computation*, volume 44. <http://dx.doi.org/10.1063/1.2810360>
- Hesse, M., Timmermann, J., Hüllermeier, E. et Trächtler, A. (2018). A reinforcement learning strategy for the swing-up of the double pendulum on a cart. *Procedia Manufacturing*, 24, 15–20. 4th International Conference on System-Integrated Intelligence: Intelligent, Flexible and Connected Systems in Products and Production, <http://dx.doi.org/https://doi.org/10.1016/j.promfg.2018.06.004>. Récupéré de <https://www.sciencedirect.com/science/article/pii/S2351978918305134>
- Ho, J., Jain, A. et Abbeel, P. (2020). Denoising diffusion probabilistic models. Dans H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, et H. Lin (dir.). *Advances in Neural Information Processing Systems*, volume 33, 6840–6851. Curran Associates, Inc. Récupéré de https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf
- Hochreiter, S. et Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9, 1735–80. <http://dx.doi.org/10.1162/neco.1997.9.8.1735>
- hoda elkhenany, abdelrahman alokda, ahmed el badawy et nagwa el badri (2018). tissue regeneration: impact of sleep on stem cell regenerative capacity. *life sciences*, 214, 51–61. <http://dx.doi.org/https://doi.org/10.1016/j.lfs.2018.10.057>. Récupéré de <https://www.sciencedirect.com/science/article/pii/S002432051830688x>
- Hoerzer, G. M., Legenstein, R. et Maass, W. (2014). Emergence of complex computational structures from chaotic neural networks through reward-modulated hebbian learning. *Cereb. Cortex*, 24(3), 677–690.
- Holca-Lamarre, R., Lücke, J. et Obermayer, K. (2017). Models of acetylcholine and dopamine signals differentially improve neural representations. *Front. Comput. Neurosci.*, 11, 54.
- Honkela, T. (1998). Learning to understand—general aspects of using self-organizing maps in natural language processing. 437. <http://dx.doi.org/10.1063/1.56323>
- Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8), 2554–2558. <http://dx.doi.org/10.1073/pnas.79.8.2554>. Récupéré de <https://www.pnas.org/content/79/8/2554>
- Hopfield, J. J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences*, 81(10), 3088–3092. <http://dx.doi.org/10.1073/pnas.81.10.3088>. Récupéré de <https://www.pnas.org/doi/abs/10.1073/pnas.81.10.3088>

- Hsu, A. L., Saeed, I. et Halgamuge, S. K. (2009). *Dynamic Self-Organising Maps: Theory, Methods and Applications*, Dans A.-E. Hassanien, A. Abraham, A. V. Vasilakos, et W. Pedrycz (dir.). *Foundations of Computational, Intelligence Volume 1: Learning and Approximation*, (p. 363–379). Springer Berlin Heidelberg: Berlin, Heidelberg
- Huang, L., Dong, X. et Clee, T. E. (2017). A scalable deep learning platform for identifying geologic features from seismic attributes. *The Leading Edge*, 36(3), 249–256.
<http://dx.doi.org/10.1190/tle36030249.1>. Récupéré de
<https://doi.org/10.1190/tle36030249.1>
- Hubel, D. H. et Wiesel, T. N. (1968). Receptive fields and functional architecture of monkey striate cortex. *J. Physiol.*, 195(1), 215–243.
- Jacques, S., Peggy, L. et Schacter, D. L. (2013). Modifying memory: selectively enhancing and updating personal memories for a museum tour by reactivating them. *Psychological science vol*, 24, 537–543. <http://dx.doi.org/10.1177/0956797612457377>
- Jones, T. A. et Adkins, D. L. (2015). Motor system reorganization after stroke: Stimulating and training toward perfection. *Physiology (Bethesda)*, 30(5), 358–370.
- Justin, M., Hubert, M. B., Betchewe, G., Doka, S. Y. et Crepin, K. T. (2019). Chaos in human brain phase transition. In P. Bracken (dir.), *Research Advances in Chaos Theory* chapitre 6. Rijeka: IntechOpen
- Kar, N. (2011). Cognitive behavioral therapy for the treatment of post-traumatic stress disorder: a review. *Neuropsychiatr. Dis. Treat.*, 7, 167–181.
- Karbowski, J. et Ermentrout, G. B. (2002). Synchrony arising from a balanced synaptic plasticity in a network of heterogeneous neural oscillators. *Phys. Rev. E*, 65, 031902.
<http://dx.doi.org/10.1103/PhysRevE.65.031902>. Récupéré de
<https://link.aps.org/doi/10.1103/PhysRevE.65.031902>
- Kargarnovin, S., Hernandez, C., Farahani, F. V. et Karwowski, W. (2023). Evidence of chaos in electroencephalogram signatures of human performance: A systematic review. *Brain Sci.*, 13(5).
- Karisetty, B. C., Bhatnagar, A., Armour, E. M., Beaver, M., Zhang, H. et Elefant, F. (2020). Amyloid-beta peptide impact on synaptic function and neuroepigenetic gene control reveal new therapeutic strategies for alzheimer's disease. *Frontiers in Molecular Neuroscience*, 13.
<http://dx.doi.org/10.3389/fnmol.2020.577622>. Récupéré de
<https://www.frontiersin.org/articles/10.3389/fnmol.2020.577622>
- katyal, s. et goldin, p. (2021). alpha and theta oscillations are inversely related to progressive levels of meditation depth. *neurosci. conscious.*, 2021(1).
- Kemker, R., McClure, M., Abitino, A., Hayes, T. et Kanan, C. (2018). Measuring catastrophic forgetting in neural networks. Dans *Proceedings of the AAAI conference on artificial intelligence*, volume 32.
- Kitzbichler, M. G., Smith, M. L., Christensen, S. R. et Bullmore, E. (2009). Broadband criticality of human brain network synchronization. *PLoS Comput. Biol.*, 5(3), e1000314.

- Knecht, S., Henningsen, H., Höhling, C., Elbert, T., Flor, H., Pantev, C. et Taub, E. (1998). Plasticity of plasticity? : Changes in the pattern of perceptual correlates of reorganization after amputation. *Brain : a journal of neurology*, 121 (Pt 4), 717–24.
<http://dx.doi.org/10.1093/brain/121.4.717>
- Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, 78(9), 1464–1480.
<http://dx.doi.org/10.1109/5.58325>
- Kohonen, T. (2013). Essentials of the self-organizing map. *Neural Networks*, 37, 52–65. Twenty-fifth Anniversary Commemorative Issue,
<http://dx.doi.org/https://doi.org/10.1016/j.neunet.2012.09.018>. Récupéré de <https://www.sciencedirect.com/science/article/pii/S0893608012002596>
- Korn, H. et Faure, P. (2003). Is there chaos in the brain? ii. experimental evidence and related models. *Comptes Rendus Biologies*, 326(9), 787–840.
<http://dx.doi.org/https://doi.org/10.1016/j.crvi.2003.09.011>. Récupéré de <https://www.sciencedirect.com/science/article/pii/S1631069103002002>
- Kosko, B. (1988). Bidirectional associative memories. *IEEE Transactions on Systems, Man, and Cybernetics*, 18(1), 49–60. <http://dx.doi.org/10.1109/21.87054>
- Krizhevsky, A., Sutskever, I. et Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Dans F. Pereira, C. J. C. Burges, L. Bottou, et K. Q. Weinberger (dir.). *Advances in Neural Information Processing Systems*, volume 25, 1097–1105. Curran Associates, Inc. Récupéré de <https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf>
- Krotov, D. et Hopfield, J. J. (2016). Dense associative memory for pattern recognition. Dans D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, et R. Garnett (dir.). *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc. Récupéré de https://proceedings.neurips.cc/paper_files/paper/2016/file/eaee339c4d89fc102edd9dbdb6a28915-Paper.pdf
- Krotov, D. et Hopfield, J. J. (2019). Unsupervised learning by competing hidden units. *Proceedings of the National Academy of Sciences*, 116(16), 7723–7731.
<http://dx.doi.org/10.1073/pnas.1820458116>. Récupéré de <https://www.pnas.org/doi/abs/10.1073/pnas.1820458116>
- Kuramoto, Y. (1975). Self-entrainment of a population of coupled non-linear oscillators. Dans H. Araki (dir.). *International Symposium on Mathematical Problems in Theoretical Physics*, 420–422., Berlin, Heidelberg. Springer Berlin Heidelberg.
- Larzabal, C., Tramoni, E., Muratot, S., Thorpe, S. J. et Barbeau, E. J. (2018). Extremely long-term memory and familiarity after 12 years. *Cognition*, 170, 254–262.
- LeCun, Y. et Cortes, C. (2005). The mnist database of handwritten digits. Récupéré le 2023 de <https://api.semanticscholar.org/CorpusID:60282629>
- Lee, H.-K. et Kirkwood, A. (2019). Mechanisms of homeostatic synaptic plasticity in vivo. *Front. Cell. Neurosci.*, 13, 520.

- Lee, S. W., O'Doherty, J. P. et Shimojo, S. (2015). Neural computations mediating one-shot learning in the human brain. *PLoS Biol.*, 13(4), e1002137.
- Liapounoff, A. (1907). Problème général de la stabilité du mouvement. *Annales de la Faculté des sciences de l'Université de Toulouse pour les sciences mathématiques et les sciences physiques*, 2e série, 9, 203–474. Récupéré de http://www.numdam.org/item/AFST_1907_2_9__203_0/
- Linnainmaa, S. (1970). *The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors*. (Master thesis). Univ. Helsinki.
- Lobo, J. L., Del Ser, J., Bifet, A. et Kasabov, N. (2020). Spiking neural networks and online learning: An overview and perspectives. *Neural Networks*, 121, 88–100.
<http://dx.doi.org/https://doi.org/10.1016/j.neunet.2019.09.004>. Récupéré de <https://www.sciencedirect.com/science/article/pii/S0893608019302655>
- Loftus, E. (1996). *Eyewitness testimony*. Cambridge: Harvard University Press.
- Lorenz, E. (1972). Does the flap of a butterfly's wings in brazil set off a tornado in texas. *139th meeting of the American Association for the Advancement of Science*.
- Lorenz, E. N. (1963). Deterministic nonperiodic flow. *J. Atmos. Sci.*, 20(2), 130–141.
- Lou, X., Fan, F., Messa, M., Raimondi, A., Wu, Y., Looger, L. L., Ferguson, S. M. et Camilli, P. D. (2012). Reduced release probability prevents vesicle depletion and transmission failure at dynamin mutant synapses. *Proceedings of the National Academy of Sciences*, 109(8), E515–E523.
<http://dx.doi.org/10.1073/pnas.1121626109>. Récupéré de <https://www.pnas.org/doi/abs/10.1073/pnas.1121626109>
- MacLeod, S., Reynolds, M. G. et Lehmann, H. (2018). The mitigating effect of repeated memory reactivations on forgetting. *npj Science of Learning*, 3(1), 9.
- Macpherson, T., Churchland, A., Sejnowski, T., DiCarlo, J., Kamitani, Y., Takahashi, H. et Hikida, T. (2021). Natural and artificial intelligence: A brief introduction to the interplay between ai and neuroscience research. *Neural Networks*, 144, 603–613.
<http://dx.doi.org/https://doi.org/10.1016/j.neunet.2021.09.018>. Récupéré de <https://www.sciencedirect.com/science/article/pii/S0893608021003683>
- Mahmoudi, B., Pohlmeier, E. A., Prins, N. W., Geng, S. et Sanchez, J. C. (2013). Towards autonomous neuroprosthetic control using hebbian reinforcement learning. *J. Neural Eng.*, 10(6), 066005.
- Makada, T., Ozair, D., Mohammed, M. et Abellanoza, C. (2016). Enhancing memory retention by increasing alpha and decreasing beta brainwaves using music.
<http://dx.doi.org/10.1145/2910674.2935851>
- Mater, A. C. et Coote, M. L. (2019). Deep learning in chemistry. *Journal of chemical information and modeling*, 59(6), 2545–2559.
- McCulloch, W. S. et Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4), 115–133.

- Millonas, M. M. (1992). A connectionist type model of self-organized foraging and emergent behavior in ant swarms. *Journal of Theoretical Biology*, 159(4), 529–552.
[http://dx.doi.org/https://doi.org/10.1016/S0022-5193\(05\)80697-6](http://dx.doi.org/https://doi.org/10.1016/S0022-5193(05)80697-6). Récupéré de <https://www.sciencedirect.com/science/article/pii/S0022519305806976>
- Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D. et Riedmiller, M. (2013). Playing atari with deep reinforcement learning.
- Moini, J. et Piran, P. (2020). Chapter 6 - cerebral cortex. In J. Moini et P. Piran (dir.), *Functional and Clinical Neuroanatomy* 177–240. Academic Press
- Molnár, Z. (2013). Chapter 7 - cortical columns. In J. L. Rubenstein et P. Rakic (dir.), *Neural Circuit Development and Function in the Brain* 109–129. Oxford: Academic Press
- Molter, C., Salihoglu, U. et Bersini, H. (2004). How chaos boosts the encoding capacity of small recurrent neural networks : learning consideration. Dans *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, volume 1, 75–80.
<http://dx.doi.org/10.1109/IJCNN.2004.1379874>
- Molter, C., Salihoglu, U. et Bersini, H. (2005). Introduction of a hebbian unsupervised learning algorithm to boost the encoding capacity of hopfield networks. Dans *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 3, 1552–1557 vol. 3.
<http://dx.doi.org/10.1109/IJCNN.2005.1556109>
- Molter, C., Salihoglu, U. et Bersini, H. (2006). How to prevent spurious data in a chaotic brain. Dans *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, 638–644.
<http://dx.doi.org/10.1109/IJCNN.2006.246743>
- Molter, C., Salihoglu, U. et Bersini, H. (2007). How stochastic noise helps memory retrieval in a chaotic brain. Dans *2007 International Joint Conference on Neural Networks*, 1458–1463.
<http://dx.doi.org/10.1109/IJCNN.2007.4371173>
- Mountcastle, V. B. (1957). Modality and topographic properties of single neurons of cat's somatic sensory cortex. *J. Neurophysiol.*, 20(4), 408–434.
- Movellan, J. R. (1991). Contrastive hebbian learning in the continuous hopfield model. In D. S. Touretzky, J. L. Elman, T. J. Sejnowski, et G. E. Hinton (dir.), *Connectionist Models* 10–17. Morgan Kaufmann
- Neelakantan, A., Vilnis, L., Le, Q. V., Sutskever, I., Kaiser, L., Kurach, K. et Martens, J. (2015). Adding gradient noise improves learning for very deep networks. *ArXiv, abs/1511.06807*. Récupéré de <https://api.semanticscholar.org/CorpusID:826188>
- Neumann, J. V., Churchland, P. M. et Neumann, K. V. (2000). *The Computer and the Brain* (2nd éd.). USA: Yale University Press.
- nina e. fultz, giorgio bonmassar, kawin setsompop, robert a. stickgold, bruce r. rosen, jonathan r. polimeni et laura d. lewis (2019). coupled electrophysiological, hemodynamic, and cerebrospinal fluid oscillations in human sleep. *science*, 366(6465), 628–631.
<http://dx.doi.org/10.1126/science.aax5440>

- Niu, D., Yang, L., Cai, T., Li, L., Wu, X. et Wang, Z. (2022). A new hierarchical temporal memory algorithm based on activation intensity. *Comput. Intell. Neurosci.*, 2022, 6072316.
- Noh, H., You, T., Mun, J. et Han, B. (2017). Regularizing deep neural networks by noise: Its interpretation and optimization. Dans I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, et R. Garnett (dir.). *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc. Récupéré de https://proceedings.neurips.cc/paper_files/paper/2017/file/217e342fc01668b10cb1188d40d3370e-Paper.pdf
- nuryadi, a., gumilar, m., lesyiana et foster, n. (2020). the impact of beta brain waves in improving cognitive function through brain jogging applications. *international journal of human movement and sports sciences*, 8. <http://dx.doi.org/10.13189/saj.2020.080713>
- O'Keefe, J. (1976). Place units in the hippocampus of the freely moving rat. *Experimental Neurology*, 51(1), 78-109. [http://dx.doi.org/https://doi.org/10.1016/0014-4886\(76\)90055-8](http://dx.doi.org/https://doi.org/10.1016/0014-4886(76)90055-8). Récupéré de <https://www.sciencedirect.com/science/article/pii/0014488676900558>
- Olvera-Cortés, M. E., Anguiano-Rodríguez, P., López-Vázquez, M. A. et Alfaro, J. M. C. (2008). Serotonin/dopamine interaction in learning. *Prog. Brain Res.*, 172, 567-602.
- OpenAI, Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., Bello, I., Berdine, J., Bernadett-Shapiro, G., Berner, C., Bogdonoff, L., Boiko, O., Boyd, M., Brakman, A.-L., Brockman, G., Brooks, T., Brundage, M., Button, K., Cai, T., Campbell, R., Cann, A., Carey, B., Carlson, C., Carmichael, R., Chan, B., Chang, C., Chantzis, F., Chen, D., Chen, S., Chen, R., Chen, J., Chen, M., Chess, B., Cho, C., Chu, C., Chung, H. W., Cummings, D., Currier, J., Dai, Y., Decareaux, C., Degry, T., Deutsch, N., Deville, D., Dhar, A., Dohan, D., Dowling, S., Dunning, S., Ecoffet, A., Eleti, A., Eloundou, T., Farhi, D., Fedus, L., Felix, N., Fishman, S. P., Forte, J., Fulford, I., Gao, L., Georges, E., Gibson, C., Goel, V., Gogineni, T., Goh, G., Gontijo-Lopes, R., Gordon, J., Grafstein, M., Gray, S., Greene, R., Gross, J., Gu, S. S., Guo, Y., Hallacy, C., Han, J., Harris, J., He, Y., Heaton, M., Heidecke, J., Hesse, C., Hickey, A., Hickey, W., Hoeschele, P., Houghton, B., Hsu, K., Hu, S., Hu, X., Huizinga, J., Jain, S., Jain, S., Jang, J., Jiang, A., Jiang, R., Jin, H., Jin, D., Jomoto, S., Jonn, B., Jun, H., Kaftan, T., Łukasz Kaiser, Kamali, A., Kanitscheider, I., Keskar, N. S., Khan, T., Kilpatrick, L., Kim, J. W., Kim, C., Kim, Y., Kirchner, J. H., Kiros, J., Knight, M., Kokotajlo, D., Łukasz Kondraciuk, Kondrich, A., Konstantinidis, A., Kosic, K., Krueger, G., Kuo, V., Lampe, M., Lan, I., Lee, T., Leike, J., Leung, J., Levy, D., Li, C. M., Lim, R., Lin, M., Lin, S., Litwin, M., Lopez, T., Lowe, R., Lue, P., Makanju, A., Malfacini, K., Manning, S., Markov, T., Markovski, Y., Martin, B., Mayer, K., Mayne, A., McGrew, B., McKinney, S. M., McLeavey, C., McMillan, P., McNeil, J., Medina, D., Mehta, A., Menick, J., Metz, L., Mishchenko, A., Mishkin, P., Monaco, V., Morikawa, E., Mossing, D., Mu, T., Murati, M., Murk, O., Mély, D., Nair, A., Nakano, R., Nayak, R., Neelakantan, A., Ngo, R., Noh, H., Ouyang, L., O'Keefe, C., Pachocki, J., Paino, A., Palermo, J., Pantuliano, A., Parascandolo, G., Parish, J., Parparita, E., Passos, A., Pavlov, M., Peng, A., Perelman, A., de Avila Belbute Peres, F., Petrov, M., de Oliveira Pinto, H. P., Michael, Pokorny, Pokrass, M., Pong, V. H., Powell, T., Power, A., Power, B., Proehl, E., Puri, R., Radford, A., Rae, J., Ramesh, A., Raymond, C., Real, F., Rimbach, K., Ross, C., Rotsted, B., Roussez, H., Ryder, N., Saltarelli, M., Sanders, T., Santurkar, S., Sastry, G., Schmidt, H., Schnurr, D., Schulman, J., Selsam, D., Sheppard, K., Sherbakov, T., Shieh, J., Shoker, S., Shyam, P., Sidor, S., Sigler, E., Simens, M., Sitkin, J., Slama, K., Sohl, I., Sokolowsky, B., Song, Y., Staudacher, N., Such, F. P., Summers, N., Sutskever, I., Tang, J.,

- Tezak, N., Thompson, M. B., Tillet, P., Tootoonchian, A., Tseng, E., Tuggle, P., Turley, N., Tworek, J., Uribe, J. F. C., Vallone, A., Vijayvergiya, A., Voss, C., Wainwright, C., Wang, J. J., Wang, A., Wang, B., Ward, J., Wei, J., Weinmann, C., Welihinda, A., Welinder, P., Weng, J., Weng, L., Wiethoff, M., Willner, D., Winter, C., Wolrich, S., Wong, H., Workman, L., Wu, S., Wu, J., Wu, M., Xiao, K., Xu, T., Yoo, S., Yu, K., Yuan, Q., Zaremba, W., Zellers, R., Zhang, C., Zhang, M., Zhao, S., Zheng, T., Zhuang, J., Zhuk, W. et Zoph, B. (2024). Gpt-4 technical report.
- Palva, J. M., Monto, S., Kulashekhar, S. et Palva, S. (2010). Neuronal synchrony reveals working memory networks and predicts individual memory capacity. *Proceedings of the National Academy of Sciences*, 107(16), 7580–7585. <http://dx.doi.org/10.1073/pnas.0913113107>. Récupéré de <https://www.pnas.org/doi/abs/10.1073/pnas.0913113107>
- Pantaleone, J. (2002). Synchronization of metronomes. *American Journal of Physics*, 70(10), 992–1000. <http://dx.doi.org/10.1119/1.1501118>. Récupéré de <https://doi.org/10.1119/1.1501118>
- Peissig, J. J. et Tarr, M. J. (2007). Visual object recognition: do we know more now than we did 20 years ago? *Annu. Rev. Psychol.*, 58(1), 75–96.
- Peters, K. Z., Cheer, J. F. et Tonini, R. (2021). Modulating the neuromodulators: Dopamine, serotonin, and the endocannabinoid system. *Trends Neurosci.*, 44(6), 464–477.
- Pickering, A. D. et Pesola, F. (2014). Modeling dopaminergic and other processes involved in learning from reward prediction error: contributions from an individual differences perspective. *Frontiers in Human Neuroscience*, 8. <http://dx.doi.org/10.3389/fnhum.2014.00740>. Récupéré de <https://www.frontiersin.org/articles/10.3389/fnhum.2014.00740>
- Preissl, H., Lutzenberger, W. et Pulvermuller, F. (1996). Is there chaos in the brain? *Behavioral and Brain Sciences*, 19(2), 307–308. <http://dx.doi.org/10.1017/S0140525X00042825>
- Ramsauer, H., Schäfl, B., Lehner, J., Seidl, P., Widrich, M., Adler, T., Gruber, L., Holzleitner, M., Pavlović, M., Sandve, G. K., Greiff, V., Kreil, D., Kopp, M., Klambauer, G., Brandstetter, J. et Hochreiter, S. (2021). Hopfield networks is all you need. Dans *Submitted to International Conference on Learning Representations*. under review. Récupéré de <https://openreview.net/forum?id=tL89RnzIiCd>
- Reis, H. J., Guatimosim, C., Paquet, M., Santos, M., Ribeiro, F. M., Kummer, A., Schenatto, G., Salgado, J. V., Vieira, L. B., Teixeira, A. L. et Palotas, A. (2009). Neuro-transmitters in the central nervous system and their implication in learning and memory processes. *Current Medicinal Chemistry*, 16(7), 796–840. <http://dx.doi.org/doi:10.2174/092986709787549271>. Récupéré de <https://www.ingentaconnect.com/content/ben/cmc/2009/00000016/00000007/art00002>
- Ribas, V. R., Ribas, R. G., Nóbrega, J. d. A., da Nóbrega, M. V., Espécie, J. A. d. A., Calafange, M. T., Calafange, C. d. O. M. et Martins, H. A. d. L. (2018). Pattern of anxiety, insecurity, fear, panic and/or phobia observed by quantitative electroencephalography (QEEG). *Dement. Neuropsychol.*, 12(3), 264–271.
- Rolls, E. T. et Treves, A. (2011). The neuronal encoding of information in the brain. *Prog. Neurobiol.*, 95(3), 448–490.

- Rosenblatt, F. (1963). *The American Journal of Psychology*, 76(4), 705–707. Récupéré le 2024-04-27 de <http://www.jstor.org/stable/1419730>
- Schirrmester, R. T., Springenberg, J. T., Fiederer, L. D. J., Glasstetter, M., Eggensperger, K., Tangermann, M., Hutter, F., Burgard, W. et Ball, T. (2017). Deep learning with convolutional neural networks for EEG decoding and visualization. *Hum. Brain Mapp.*, 38(11), 5391–5420.
- Shah-Hosseini, H. et Safabakhsh, R. (2000). Tasom: the time adaptive self-organizing map. 422–427. <http://dx.doi.org/10.1109/ITCC.2000.844265>
- Shahamat, H. et Saniee Abadeh, M. (2020). Brain mri analysis using a deep learning based evolutionary approach. *Neural Networks*, 126, 218–234. <http://dx.doi.org/https://doi.org/10.1016/j.neunet.2020.03.017>. Récupéré de <https://www.sciencedirect.com/science/article/pii/S0893608020300940>
- Shaw, J. et Porter, S. (2015). Constructing rich false memories of committing crime. *Psychological science*, 26. <http://dx.doi.org/10.1177/0956797614562862>
- Siestrup, S. et Schubotz, R. I. (2023). Minor Changes Change Memories: Functional Magnetic Resonance Imaging and Behavioral Reflections of Episodic Prediction Errors. *Journal of Cognitive Neuroscience*, 35(11), 1823–1845. http://dx.doi.org/10.1162/jocn_a_02047. Récupéré de https://doi.org/10.1162/jocn_a_02047
- Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T. et al. (2017). Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*.
- Skolnick, J., Jones, W., Alasoo, K., Fishman, D. et Parts, L. (2017). Computational biology: deep learning. *Emerging Topics in Life Sciences*, 1(3), 257–274. <http://dx.doi.org/10.1042/ETLS20160025>. Récupéré de <https://doi.org/10.1042/ETLS20160025>
- Skupin, A., Biberstine, J. R. et Börner, K. (2013). Visualizing the topical structure of the medical sciences: A self-organizing map approach. *PLOS ONE*, 8(3), 1–16. <http://dx.doi.org/10.1371/journal.pone.0058779>. Récupéré de <https://doi.org/10.1371/journal.pone.0058779>
- Song, S., Miller, K. D. et Abbott, L. F. (2000). Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nature Neuroscience*, 3(9), 919–926.
- Steyn-Ross, M., Steyn-Ross, A. et Sleigh, J. (2014). The sleeping brain regulates to the edge of chaos. *BMC Neuroscience*, 15, O19–O19. <http://dx.doi.org/10.1186/1471-2202-15-S1-O19>
- Storkey, A. (1997). Increasing the capacity of a hopfield network without sacrificing functionality. Dans W. Gerstner, A. Germond, M. Hasler, et J.-D. Nicoud (dir.). *Artificial Neural Networks — ICANN'97*, 451–456., Berlin, Heidelberg. Springer Berlin Heidelberg.
- Storkey, A. (1999). Increasing the capacity of a hopfield network without sacrificing functionality. 1327. <http://dx.doi.org/10.1007/BFb0020196>

- Strausfeld, N., Ma, X. et Edgecombe, G. (2016). Fossils and the evolution of the arthropod brain. *Current Biology*, 26(20), R989–R1000.
<http://dx.doi.org/https://doi.org/10.1016/j.cub.2016.09.012>. Récupéré de <https://www.sciencedirect.com/science/article/pii/S0960982216310612>
- Tabas, A. et von Kriegstein, K. (2021). Neural modelling of the encoding of fast frequency modulation. *PLoS Comput. Biol.*, 17(3), e1008787.
- Talaván, P. M. et Yáñez, J. (2005). A continuous hopfield network equilibrium points algorithm. *Computers and Operations Research*, 32(8), 2179–2196.
<http://dx.doi.org/https://doi.org/10.1016/j.cor.2004.02.008>. Récupéré de <https://www.sciencedirect.com/science/article/pii/S0305054804000243>
- tang, y.-y., tang, r., rothbart, m. k. et posner, m. i. (2019). frontal theta activity and white matter plasticity following mindfulness meditation. *curr. opin. psychol.*, 28, 294–297.
- Tellez, R., Gómez-Viquez, L., Liy-Salmeron, G. et Meneses, A. (2012a). Gaba, glutamate, dopamine and serotonin transporters expression on forgetting. *Neurobiology of Learning and Memory*, 98(1), 66–77. <http://dx.doi.org/https://doi.org/10.1016/j.nlm.2012.05.001>. Récupéré de <https://www.sciencedirect.com/science/article/pii/S107474271200069X>
- Tellez, R., Gómez-Viquez, L. et Meneses, A. (2012b). Gaba, glutamate, dopamine and serotonin transporters expression on memory formation and amnesia. *Neurobiology of Learning and Memory*, 97(2), 189–201.
<http://dx.doi.org/https://doi.org/10.1016/j.nlm.2011.12.002>. Récupéré de <https://www.sciencedirect.com/science/article/pii/S1074742711002012>
- Thuerey, N., Holl, P., Mueller, M., Schnell, P., Trost, F. et Um, K. (2022). Physics-based deep learning. Récupéré de <https://arxiv.org/abs/2109.05237>
- Tiedemann, H., Morgenstern, Y., Schmidt, F. et Fleming, R. W. (2022). One-shot generalization in humans revealed through a drawing task. *eLife*, 11, e75485.
<http://dx.doi.org/10.7554/eLife.75485>. Récupéré de <https://doi.org/10.7554/eLife.75485>
- Tomasello, R., Garagnani, M., Wennekers, T. et Pulvermüller, F. (2018). A neurobiologically constrained cortex model of semantic grounding with spiking neurons and brain-like connectivity. *Frontiers in Computational Neuroscience*, 12. <http://dx.doi.org/10.3389/fncom.2018.00088>. Récupéré de <https://www.frontiersin.org/articles/10.3389/fncom.2018.00088>
- tsanov, m. et manahan vaughan, d. (2009). long-term plasticity is proportional to theta-activity. *plos one*, 4(6), e5850.
- Turing, A. (2004). 395Intelligent Machinery (1948). In *The Essential Turing*. Oxford University Press
- Turner, G. et Onysk, J. (2022). The hippocampus may support context retrieval in one-shot learning about pain. *Journal of Neuroscience*, 42(10), 1883–1885.
<http://dx.doi.org/10.1523/JNEUROSCI.2250-21.2022>. Récupéré de <https://www.jneurosci.org/content/42/10/1883>

- Uddén, J., Folia, V. et Petersson, K. M. (2010). The neuropharmacology of implicit learning. *Curr. Neuropharmacol.*, 8(4), 367–381.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, u. et Polosukhin, I. (2017). Attention is all you need. Dans *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17*, p. 6000–6010., Red Hook, NY, USA. Curran Associates Inc.
- Volterra, V. (1926). Fluctuations in the abundance of a species considered mathematically. *Nature*, 118, 558–560. <http://dx.doi.org/10.1038/118558a0>
- Wert-Carvajal, C., Reneaux, M., Tchumatchenko, T. et Clopath, C. (2022). Dopamine and serotonin interplay for valence-based spatial learning. *Cell Rep.*, 39(2), 110645.
- Whittington, J. C. R., Warren, J. et Behrens, T. E. J. (2022). Relating transformers to models and neural representations of the hippocampal formation.
- Williams, L. E. et Bargh, J. A. (2008). Experiencing physical warmth promotes interpersonal warmth. *Science*, 322(5901), 606–607.
- Witter, M. (2012). Chapter 5 - hippocampus. In C. Watson, G. Paxinos, et L. Puelles (dir.), *The Mouse Nervous System* 112–139. San Diego: Academic Press
- Wixted, J. T., Squire, L. R., Jang, Y., Papesh, M. H., Goldinger, S. D., Kuhn, J. R., Smith, K. A., Treiman, D. M. et Steinmetz, P. N. (2014). Sparse and distributed coding of episodic memory in neurons of the human hippocampus. *Proceedings of the National Academy of Sciences*, 111(26), 9621–9626. <http://dx.doi.org/10.1073/pnas.1408365111>. Récupéré de <https://www.pnas.org/doi/abs/10.1073/pnas.1408365111>
- Xing, F., Xie, Y., Su, H., Liu, F. et Yang, L. (2018). Deep learning in microscopy image analysis: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 29(10), 4550–4568. <http://dx.doi.org/10.1109/TNNLS.2017.2766168>
- Yin, X. et Wang, R. (2016). Simulation of dopamine modulation-based memory model. *Neurocomputing*, 194, 241–245. <http://dx.doi.org/https://doi.org/10.1016/j.neucom.2016.01.077>. Récupéré de <https://www.sciencedirect.com/science/article/pii/S0925231216002381>
- Yu, L., Shen, Z., Wang, C. et Yu, Y. (2018). Efficient coding and energy efficiency are promoted by balanced excitatory and inhibitory synaptic currents in neuronal network. *Front. Cell. Neurosci.*, 12.
- Zhang, J., Rao, V. M., Tian, Y., Yang, Y., Acosta, N., Wan, Z., Lee, P.-Y., Zhang, C., Kegeles, L. S., Small, S. A. et Guo, J. (2023). Detecting schizophrenia with 3D structural brain MRI using deep learning. *Sci. Rep.*, 13(1), 14433.
- Zhang, X., Yao, L., Wang, X., Monaghan, J., Mcalpine, D. et Zhang, Y. (2019). A survey on deep learning based brain computer interface: Recent advances and new frontiers. *arXiv preprint arXiv:1905.04149*, 66.