



HAL
open science

Étude probabiliste de modèles neuronaux de mémoire associative et d'algorithmes utilisés en physique statistique et data science

Franck Vermet

► **To cite this version:**

Franck Vermet. Étude probabiliste de modèles neuronaux de mémoire associative et d'algorithmes utilisés en physique statistique et data science. Mathématiques [math]. Université de Bretagne occidentale; Ecole doctorale MathSTIC, 2019. tel-04949466

HAL Id: tel-04949466

<https://hal.science/tel-04949466v1>

Submitted on 15 Feb 2025

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

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

Habilitation à diriger des recherches

Spécialité : Mathématiques appliquées et applications des mathématiques

présentée

DEVANT L'UNIVERSITÉ DE BREST

par

Franck VERMET

UFR Sciences et Techniques

Laboratoire de Mathématiques de Bretagne Atlantique, UMR 6205

Réseau doctoral Ouest Mathématiques

École doctorale MathSTIC

TITRE DES TRAVAUX :

*Étude probabiliste de modèles neuronaux de mémoire
associative et d'algorithmes utilisés en physique statistique
et data science*

Soutenue le 29 mai 2019 devant le jury composé de :

- | | | |
|-----|-------------------|--|
| M. | BOIVIN Daniel | Professeur des universités, Université de Brest |
| M. | FRANKE Brice | Professeur des universités, Université de Brest |
| M. | LE NY Arnaud | Professeur des universités, Université Paris-Est
Créteil Val-de-Marne |
| Mme | LOUKIANOVA Dasha | Maître de Conférences, Université d'Evry |
| M. | PETRITIS Dimitri | Professeur des universités, Université de Rennes 1 |
| M. | VAN ENTER Aernout | Professeur des universités, Université de Groningue (Pays-Bas) |

Au vu des rapports de :

- | | | |
|-----|---------------------|--|
| M. | EICHELSBACHER Peter | Professeur des universités, Université de Bochum (Allemagne) |
| Mme | LOUKIANOVA Dasha | Maître de Conférences, Université d'Evry |
| M. | VAN ENTER Aernout | Professeur des universités, Université de Groningue (Pays-Bas) |

*A la mémoire d’Azeline et Céline,
qui nous ont montré les valeurs essentielles.*

Je tiens à remercier de nombreuses personnes qui ont permis la réalisation des travaux présentés dans ce document.

Je tiens tout d’abord à remercier chaleureusement Dimitri Petritis, qui m’a fait découvrir en DEA le monde passionnant de la physique mathématique et m’a proposé un sujet de thèse sur le modèle de réseau de neurones de Hopfield. Ce fut un grand bonheur de débiter mes activités de recherche avec lui et toute ma carrière d’enseignant-chercheur a été largement influencée par ces premières années.

Je remercie également tout particulièrement Matthias Löwe avec qui j’ai partagé la majorité de mes projets de recherche, avec un plaisir et une bonne humeur toujours partagés. Que de bons moments passés à travailler ensemble !

Merci également à mes autres collaborateurs, Pierre Ailliot, Mete Demircigil, Mirko Ebbers, Darran Furnival, Vincent Gripon, Ghouthi Hacene, Judith Heusel, Remco Van der Hofstad, Holger Knöpfel, Robert Scott, Sven Upgang et Corwin Wright, ainsi qu’à mes collègues de travail à l’Université de Brest.

J’ai une pensée également pour les étudiantes et étudiants de l’EURIA, aujourd’hui actives, que j’ai encadrés au cours de différents projets, de bureaux d’études ou pour la réalisation de leur mémoire. Une pensée particulière pour Aissatou Dieng, Korotoumou Traoré et Fatima-Zohra Zougagh avec qui j’ai beaucoup échangé. C’est en partie ce travail d’encadrement passionnant qui m’a donné envie d’obtenir mon habilitation à diriger des recherches pour pouvoir encadrer des thèses.

Enfin, je remercie ma famille, et tout particulièrement ma femme Pauline et mes enfants Adrien et Jovana qui éclairent mon chemin de bonheur quotidien.

Table des matières

Introduction	3
Première partie 1. Sur la capacité de stockage des modèles neuronaux de mémoire associative	5
1. Le modèle de Hopfield	6
2. Autres modèles de mémoire associative	15
Deuxième partie 2. Liens entre le modèle de Hopfield et le système CDMA	25
3. Le modèle CDMA	26
4. Application au modèle de Hopfield	29
5. Un résultat de déviations modérées	30
Troisième partie 3. Etude d'un algorithme stochastique : la méthode du swapping	33
6. Les algorithmes de simulated tempering et de swapping	34
7. Application au modèle de Hopfield	37
8. Application au modèle de Blume-Emery-Griffiths	38
Quatrième partie 4. Etude probabiliste d'un algorithme de classification lié à l'apprentissage profond	41
9. Un algorithme de classification avec segmentation	43
10. Une situation où la segmentation n'est pas bénéfique	43
11. Des situations où la segmentation est bénéfique	44
Travaux de l'auteur	49
Bibliographie	51

Introduction

Ce document est une synthèse de mes travaux de recherche. Je vais présenter les résultats obtenus et publiés depuis 2005. Mon intérêt pour le modèle de mémoire associative de Hopfield, qui a fait l'objet de ma thèse de doctorat, a été le point de départ de la majorité des articles que j'ai publiés avec mes coauteurs.

J'ai regroupé mes travaux en quatre parties, selon les thèmes de recherche abordés. Dans chacune de ces parties, je présente les modèles étudiés et une synthèse des articles publiés. Les démonstrations peuvent être consultées dans les articles cités, joints à ce document.

Dans la première partie, je présente les résultats sur des extensions du modèle de Hopfield. Les travaux portent sur la capacité de stockage, qui est influencée par les caractéristiques du modèle : loi et propriétés de dépendance des messages mémorisés, structure du graphe modélisant le réseau de neurones. Nous nous sommes également beaucoup intéressés avec mes coauteurs à la capacité de stockage d'autres modèles de mémoire associative, les modèles d'Amari, de Willshaw, de Gripon et Berrou, de Hopfield et Krotov, et nous avons proposé des variantes pour ces modèles et leur dynamique.

La seconde partie est consacrée au système de télécommunication CDMA, pour lequel nous étudions la capacité (i.e. le nombre maximal d'utilisateurs possible) en fonction des algorithmes de décodage utilisés. Nous nous sommes en particulier intéressés aux algorithmes de décodage par filtre adapté et aux méthodes de suppression d'interférences HD-PIC et SD-PIC. Nous avons mis en évidence une analogie entre ce modèle et le modèle de Hopfield, ainsi qu'avec une variante du modèle de Hopfield proposée par Amari et Yanai. Grâce à un principe de déviations modérées pour des sommes de variables faiblement dépendantes que nous avons établi, nous avons démontré des résultats annoncés par Amari et Yanai pour leur modèle de mémoire associative et nous avons pu analyser le comportement du modèle CDMA dans le cas de l'application d'une étape de l'algorithme de suppression d'interférences SD-PIC.

Dans la troisième partie, nous nous intéressons aux algorithmes stochastiques de type Monte Carlo par chaîne de Markov. L'algorithme de Metropolis-Hastings est utilisé notamment en physique mathématique pour générer des échantillons de lois de probabilité complexes, mais peut avoir un temps de convergence exponentiel. Des variantes ont été proposées pour accélérer la convergence, dont les méthodes de *swapping* et *simulated tempering*, et nous nous sommes intéressés à ces algorithmes dans le cas du modèle de Hopfield. Nous avons pu traiter le cas où le modèle stocke deux images, généralisant ainsi le cas connu du modèle de Curie Weiss à un modèle désordonné en montrant un temps de convergence polynomial pour l'algorithme du *swapping*. Nous avons également étudié ces algorithmes dans le cas du modèle de Blume-Emery-Griffiths et mis en évidence sur

cet exemple l'influence de la nature des transitions de phase du modèle sur la vitesse de convergence des algorithmes.

Enfin, la dernière partie de cette synthèse concerne des travaux récents en lien avec les réseaux de neurones à apprentissage profond. Le modèle de Hopfield m'a naturellement amené à m'intéresser à d'autres types de réseaux de neurones, en particulier les réseaux de neurones structurés en couches qui sont utilisés en apprentissage statistique supervisé pour des applications en régression non linéaire et en classification. Ainsi, dans le cadre de travaux communs avec Matthias Löwe et Vincent Gripon, nous avons proposé une méthode de classification adaptée aux réseaux de neurones de type *Deep learning* dans le cadre de technique de transfert d'apprentissage.

Première partie

Sur la capacité de stockage des modèles neuronaux de mémoire associative

Cette partie correspond aux articles [7], [9], [12], [15], [16], [17] et [18] des travaux joints. Il s'agit de travaux portant sur la capacité de stockage de modèles neuronaux de mémoire associative. Dans un premier temps, il s'agit de résultats généralisant les résultats connus pour le modèle de Hopfield, obtenus en modifiant les hypothèses originales du modèle.

Dans la deuxième section, nous considérons d'autres modèles, notamment basés sur les cliques neurales.

1. Le modèle de Hopfield

Introduisons le formalisme du modèle de Hopfield (1982) [**Hopf**]. Nous considérons N neurones connectés selon le graphe complet $G = K_N$. Chaque neurone peut être dans 2 états : -1 ou $+1$ et l'espace des configurations du réseau est alors $S = \{-1, +1\}^N$.

L'information à mémoriser est constituée de M messages binaires de longueur N : $(\xi^\mu)_{\mu=1}^M = (\xi_1^\mu, \dots, \xi_N^\mu)_{\mu=1}^M$, avec $\xi_i^\mu = \pm 1$. Les ξ_i^μ sont des variables aléatoires définies sur un espace probabilisé $(\Omega, \mathcal{F}, \mathbb{P})$, indépendantes et de loi uniforme sur $\{-1, +1\}$.

Les poids synaptiques contiennent l'information : pour $i, j = 1, \dots, N, i \neq j$,

$$J_{ij} = \sum_{\mu=1}^M \xi_i^\mu \xi_j^\mu$$

Nous définissons une application sur l'espace S des configurations

$$\begin{aligned} T : S &\longrightarrow S \\ \sigma &\longmapsto T(\sigma) = (T_i(\sigma))_{i=1, \dots, N}, \quad T_i(\sigma) = \operatorname{sgn}\left(\sum_{\substack{j=1 \\ j \neq i}}^N J_{ij} \sigma_j\right) \end{aligned}$$

La mémoire associative est alors induite par une dynamique parallèle sur S . L'état des neurones évolue au cours du temps : pour $\sigma(0) \in S$,

$$\sigma(t+1) = T(\sigma(t)), \quad t \in \mathbb{N}.$$

Il est facile de montrer la propriété suivante : pour tout $\sigma(0) \in S$, la dynamique converge soit vers une configuration stable σ^* , i.e. $T(\sigma^*) = \sigma^*$, soit vers un 2-cycle $(\tilde{\sigma}^1, \tilde{\sigma}^2) \in S^2$, i.e. $T(\tilde{\sigma}^1) = \tilde{\sigma}^2, T(\tilde{\sigma}^2) = \tilde{\sigma}^1$.

Le principe de *mémoire associative sans erreurs* que nous espérons est le suivant :

- Les M messages mémorisés sont stables : $T(\xi^\mu) = \xi^\mu, \forall \mu = 1, \dots, M$.

- Le système peut corriger des « erreurs » et retrouver les messages mémorisés à partir d'une connaissance partielle de ces messages : si $\sigma(0) \sim \xi^\mu$, alors $T^k(\sigma(0))$ converge vers ξ^μ (correction des erreurs en k étapes de la dynamique).

Il est possible d'obtenir des résultats asymptotiques dans ce cadre probabiliste, en considérant $N \rightarrow +\infty$ et donc aussi $M \rightarrow +\infty$ pour que cela soit intéressant. Une question naturelle porte sur la capacité de stockage du réseau : quel est le nombre maximal de messages que le système puisse mémoriser ? La mémorisation correspond ici à la stabilité des messages appris pour la dynamique et à la capacité de corriger des erreurs à partir de versions bruitées de ces dernières.

Nous allons mentionner les résultats classiques pour le modèle de Hopfield, avant de présenter nos travaux sur ce sujet.

THÉORÈME 1 (stabilité des messages appris). *Soit $M = c \frac{N}{\log N}$. Alors :*

- (Mac Eliece et al, 1987) [**MacE**]
- Si $c < 1/2$, alors pour tout μ ,

$$\mathbb{P}(T(\xi^\mu) = \xi^\mu) \geq 1 - N^{1-1/2c} \rightarrow 1 \text{ (lorsque } N \rightarrow \infty)$$

- Si $c < 1/4$, alors

$$\mathbb{P}(\forall \mu = 1, \dots, M, T(\xi^\mu) = \xi^\mu) \rightarrow 1.$$

- (Bovier, 1998) [**Bov**]
Si $c > 1/2$, alors pour tout μ ,

$$\mathbb{P}(T(\xi^\mu) = \xi^\mu) \rightarrow 0.$$

Soit d_H la distance de Hamming sur $S = \{-1, +1\}^N$, i.e. $d_H(x, y)$ est le nombre de coordonnées où x and y diffèrent, et

$$\mathcal{S}(\xi^\mu, \rho) = \{x \in S : d_H(\xi^\mu, x) = \rho N\}, \text{ pour } \mu = 1, \dots, M.$$

THÉORÈME 2 (correction des erreurs). *Soit $M = c \frac{N}{\log N}$. Alors :*

- (Komlos, Paturi, 1988) [**KP1**]
Si $c < 1/4$, alors il existe $\rho_0 \in]0, 1/2[$, tel que pour tout $\rho \in [0, \rho_0[$, il existe $k = \mathcal{O}(\log \log(N))$, tel que

$$\mathbb{P}(\forall \mu = 1, \dots, M, \forall y \in \mathcal{S}(\xi^\mu, \rho), T^k(y) = \xi^\mu) \rightarrow 1.$$

- (Burshtein, 1994) [**Bur**]
Si $c < 1/2$, alors pour tout μ , pour tout $\rho \in [0, 1/2[$, et toute configuration y choisie au hasard dans $\mathcal{S}(\xi^\mu, \rho)$, il existe un k (dépendant de ρ , mais pas de N), tel que

$$\mathbb{P}(T^k(y) = \xi^\mu) \rightarrow 1 \text{ (lorsque } N \rightarrow \infty)$$

Avec Matthias Löwe, nous avons proposé des variantes du modèle de Hopfield et obtenu des résultats sur la capacité de stockage dans un cadre moins restrictif que celui proposé par J. Hopfield.

Dans l'article [7], nous levons la contrainte d'indépendance des v.a. ξ_i^μ en autorisant des dépendances entre les coordonnées d'un même message, ou bien entre les coordonnées de différents messages.

THÉORÈME 3 ([7]). *Supposons que les ξ_i^μ sont des variables aléatoires de loi uniforme sur $\{-1, +1\}$ et que pour tous i et μ , la fonction de « corrélation »*

$$X_N^{i,\mu} = \frac{1}{N} \sum_{\substack{1 \leq j \leq N \\ j \neq i}} \sum_{\substack{1 \leq \nu \leq M \\ \nu \neq \mu}} \xi_i^\mu \xi_j^\mu \xi_i^\nu \xi_j^\nu$$

vérifie une borne supérieure de déviation modérée avec vitesse $\frac{N^\alpha}{M^\beta}$, pour certains $\alpha, \beta > 0$ et fonction de taux $I(\cdot)$, telle que $J(-1) = \inf_{x \leq -1} I(x) > 0$, alors nous avons les bornes suivantes pour la capacité de stockage du modèle de Hopfield. Supposons que $M = \frac{N^{\alpha/\beta}}{\gamma(\log N)^{1/\beta}}$. Alors :

$$(1) \text{ Si } \gamma > \left(\frac{3}{J(-1)}\right)^{1/\beta},$$

$$\mathbb{P}(\liminf_{N \rightarrow \infty} (\cap_{\mu=1}^M T\xi^\mu = \xi^\mu)) = 1.$$

$$(2) \text{ Si } \gamma \geq \left(\frac{2}{J(-1)}\right)^{1/\beta},$$

$$\mathbb{P}((\cap_{\mu=1}^M T\xi^\mu = \xi^\mu)) = 1 - R_N \quad \text{avec } \lim_{N \rightarrow \infty} R_N = 0.$$

$$(3) \text{ Si } \gamma > \left(\frac{1}{J(-1)}\right)^{1/\beta}, \text{ pour chaque } \mu = 1, \dots, M,$$

$$\mathbb{P}(T\xi^\mu = \xi^\mu) = 1 - R_N \quad \text{avec } \lim_{N \rightarrow \infty} R_N = 0.$$

Ce cadre permet par exemple d'obtenir des bornes explicites dans le cas de dépendances Markoviennes entre les v.a. ξ_i^μ (spatiales ou temporelles), ainsi que de traiter des cas où les messages sont des images modélisées par des champs aléatoires (voir [7]).

Dans l'article [9], nous avons étudié la capacité du modèle de Hopfield à q états possibles (appelé aussi modèle neuronal de Potts-Hopfield). Nous avons choisi comme espace des états possibles pour chaque coordonnée d'un message

$$I = \{v_i = e_i - 1/q\mathbf{1}, i = 1, \dots, q\},$$

où e_i est le $i^{\text{ème}}$ vecteur unité dans \mathbb{R}^q et $\mathbf{1} \in \mathbb{R}^q$ est le vecteur avec chaque coordonnée égale à 1. Nous considérons M vecteurs $\xi^1, \dots, \xi^M \in S = I^N$ à mémoriser et définissons

$$W_{ij} = \sum_{\mu=1}^M \xi_i^\mu (\xi_j^\mu)^t$$

où l'indice t correspond à la transposée. Ainsi chaque W_{ij} est une matrice $q \times q$. Pour chaque $\sigma \in S$, la dynamique $T = (T_i)_{i=1}^N$ est alors définie par

$$T_i(\sigma) = \pi \left(\sum_{j=1}^N W_{ij} \sigma_j \right),$$

où $W_{ij} \sigma_j$ est un produit de matrices et π la projection sur S dans \mathbb{L}^2 (i.e. la projection qui minimise la distance Euclidienne). Pour $q = 2$ nous retrouvons un modèle équivalent au modèle de Hopfield standard.

Nous supposons que les $\xi^1, \dots, \xi^M \in S$ sont indépendants de loi uniforme sur S , i.e. les vecteurs ξ^μ sont indépendants et les coordonnées de chaque vecteur sont choisies indépendamment et avec probabilité uniforme dans I .

Nous avons démontré les résultats suivants sur la capacité de stockage du modèle.

THÉORÈME 4 ([9]). Soit $M = c \frac{N}{\log N}$. Alors

(1) Si $c < \frac{q(q-1)}{4}$, alors pour chaque $\mu = 1, \dots, M$ fixé,

$$\mathbb{P}(T(\xi^\mu) = \xi^\mu) \rightarrow 1 \quad \text{lorsque } N \rightarrow \infty.$$

(2) Si $c < \frac{q(q-1)}{8}$, alors

$$\mathbb{P}(\forall \mu = 1, \dots, M : T(\xi^\mu) = \xi^\mu) \rightarrow 1 \quad \text{lorsque } N \rightarrow \infty.$$

(3) Si $c < \frac{q(q-1)}{12}$, alors

$$\mathbb{P}(\liminf_{N \rightarrow \infty} \bigcap_{\mu=1}^M T(\xi^\mu) = \xi^\mu) = 1.$$

Pour $q = 2$ nous retrouvons les résultats connus concernant le modèle de Hopfield original (voir [MacE], [3]). Remarquons que la capacité augmente avec q , la capacité étant proportionnelle à $q(q-1)$.

Le théorème suivant généralise le résultat de Bovier ([Bov]) pour le modèle de Hopfield standard et montre que la borne obtenue dans la partie (1) du théorème 4 est optimale.

THÉORÈME 5 ([9]). Soit $M = c \frac{N}{\log N}$. Alors, si $c > \frac{q(q-1)}{4}$,

$$\mathbb{P}(T(\xi^\mu) = \xi^\mu) \rightarrow 0 \quad \text{lorsque } N \rightarrow \infty \text{ pour tous } \mu = 1, \dots, M.$$

Nous avons aussi obtenu des résultats à propos de la correction d'erreurs de messages corrompus, qui est la propriété de mémoire associative souhaitée.

THÉORÈME 6 ([9]). Soit $M = c \frac{N}{\log N}$. Alors si $\varrho < \frac{q-1}{q}$ et $c < \frac{q^2(1-\frac{1}{q}-\varrho)^2}{4(1-\frac{1}{q})}$, pour tout μ et tout message y^μ choisi au hasard dans $\mathcal{S}_{\varrho N}(\xi^\mu) = \{x \in S^N, d_H(x, \xi^\mu) = \varrho N\}$, où $d_H(\cdot, \cdot)$ est la distance de Hamming et nous supposons que ϱN est un entier, nous avons :

$$\mathbb{P}(T(y^\mu) = \xi^\mu) \rightarrow 1 \quad \text{as } N \rightarrow \infty.$$

REMARQUE 7. Des résultats analogues aux parties (2) et (3) du théorème 4 sont vrais également pour la correction d'erreurs. Les valeurs correspondantes des constantes sont respectivement $c < \frac{q^2(1-\frac{1}{q}-\varrho)^2}{8(1-\frac{1}{q})}$ et $c < \frac{q^2(1-\frac{1}{q}-\varrho)^2}{12(1-\frac{1}{q})}$.

Le théorème 6 traite le cas de la correction des erreurs en une étape de la dynamique. On peut alors se demander si pour M plus grand, i.e. $M = c \frac{N}{\log N}$ avec $c < q(q-1)/4$ arbitraire, la dynamique peut corriger des erreurs, éventuellement en plusieurs étapes. Le théorème suivant répond à cette question.

THÉORÈME 8 ([9]). Soit $M = c \frac{N}{\log N}$. Alors, si $c < q(q-1)/4$, pour tout $\mu = 1, \dots, M$, tout $0 \leq \varrho < \frac{q-1}{q}$ et tout y^μ choisi au hasard dans $\mathcal{S}_{\varrho N}(\xi^\mu)$, il existe un k (dépendant de ϱ) tel que

$$\mathbb{P}(T^k(y^\mu) = \xi^\mu) \rightarrow 1$$

lorsque $N \rightarrow \infty$. Ici T^k est l'itérée k fois de l'application T .

Dans les articles [12] et [15], Matthias Löwe et moi avons considéré le cas où le modèle de Hopfield n'est plus défini sur un graphe complet, c'est-à-dire que les neurones ne sont pas tous interconnectés. Donnons le formalisme mathématique correspondant à ce cas. Soit $G = (V = \{1, \dots, N\}, E)$ un graphe non orienté. Soit A la matrice symétrique d'adjacence associée à G :

$$A_{ij} = \begin{cases} 1, & \text{si } \{i, j\} \in E \\ 0, & \text{sinon} \end{cases}$$

Dans le cas où le modèle de Hopfield est défini sur G , seuls les liens du réseau portent une information :

$$J_{ij} = \sum_{\mu=1}^M \xi_i^\mu \xi_j^\mu, \text{ pour } \{i, j\} \in E.$$

La dynamique parallèle devient alors : $T(\sigma) = (T_i(\sigma))$ avec

$$T_i(\sigma) = \operatorname{sgn}\left(\sum_{\{i,j\} \in E} J_{ij} \sigma_j\right) = \operatorname{sgn}\left(\sum_{i,j=1}^N A_{ij} J_{ij} \sigma_j\right) \quad (1)$$

Les questions que l'on se pose sont :

- Le modèle de Hopfield peut-il fonctionner correctement sur un réseau arbitraire ?
- Quelle est l'incidence du graphe sous-jacent sur la capacité de stockage du modèle ?

Nous avons montré que la capacité du modèle est liée aux propriétés spectrales de la matrice d'adjacence, une propriété qui avait déjà été mise en évidence par Komlos et Paturi dans le cas des graphes réguliers (voir [KP2]). Résumons les propriétés importantes :

- Les valeurs propres de la matrice A sont réelles :

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N \quad \text{et} \quad \lambda_1 \geq \kappa = \max_{i \geq 2} |\lambda_i|.$$

- Les valeurs propres de A donnent une information sur la connectivité du graphe :
 - si le graphe est connexe, alors λ_1 est de multiplicité 1.
 - degré moyen $\leq \lambda_1 \leq$ degré maximal.
 - l'écart entre λ_1 et κ « mesure » la connectivité du graphe. Par exemple,
 - si $G = K_N$, le graphe complet, alors $\lambda_1 = N - 1$ et $\lambda_i = -1$, pour $i \geq 2$.
 - si G est régulier et non connexe, alors $\lambda_1 = \lambda_2 = \kappa$.

Nous avons montré le résultat général suivant. Soient $d_i = \sum_{j=1}^N A_{ij}$ le degré du vertex

i , $\delta = \min_i d_i$, et $m = \max_i d_i$.

THÉORÈME 9 ([15]). *S'il existe $c_1 \in]0, 1[$ et $c_2 > 0$ suffisamment grand tel que*

$$\delta > c_1 \lambda_1 \quad \text{et} \quad \lambda_1 \geq c_2 \log(N) \kappa,$$

alors il existe $\alpha_0 > 0$ et $\rho_0 \in]0, 1/2[$ tel que si

$$M = \alpha \frac{\lambda_1^2}{m \log N} - \frac{\kappa \lambda_1}{m},$$

avec $\alpha < \alpha_0$, alors pour tout $\rho \in]0, \rho_0]$ nous avons

$$\mathbb{P}[\forall \mu = 1, \dots, M, \forall x \in \mathcal{S}(\xi^\mu, \rho) : T^k(x) = \xi^\mu] \rightarrow 1, \text{ as } N \rightarrow \infty,$$

pour $k = \mathcal{O}(\max\{\log \log N, \frac{\log(N)}{\log(\frac{\lambda_1}{\kappa \log(N)})}\})$.

Donnons quelques exemples explicites :

- Si G est d -régulier, alors $\lambda_1 = m = d$ et nous retrouvons un résultat de Komlos et Paturi (1993) ([**KP2**]), avec une capacité d'ordre $d/\log(N)$, si $\lambda_1 \geq c \log(N) \kappa$ (c assez grand).
- Si G est un graphe de Erdős-Renyi $G(N, p)$ de paramètre $p \geq c \frac{(\log N)^2}{N}$, la capacité est d'ordre $pN/\log(N)$.

En effet, dans ce cas, avec probabilité $\rightarrow 1$, le graphe est connexe et

$$\lambda_1 = (1 + o(1))pN, \quad \kappa \leq c\sqrt{Np},$$

$$m = (1 + o(1))pN, \quad \delta = (1 + o(1))pN.$$

- Si G est un graphe de type « power law » ou « scale free » (il s'agit de graphes dont le nombre de sommets de degré k est proportionnel à $k^{-\beta}$, pour un certain paramètre $\beta > 2$), la capacité est d'ordre

$$\frac{d^2}{m \log(N)}$$

pour le modèle de Chung et Lu (2002) ([**CL**]), dans le cas où $\beta > 3$, et

$$m \gg d > c\sqrt{m}(\log(N))^{3/2}.$$

Ici, d est le degré moyen du graphe. Ces résultats sont obtenues en utilisant des résultats de Chung et Radcliffe (2011) ([**CR**]) sur le spectre des matrices d'adjacence de certains graphes de ce type.

Jusqu'à présent, nous nous sommes intéressés à la correction sans erreur des messages corrompus, c'est-à-dire que nous considérons le cas où l'itération de l'application T corrige toutes les erreurs et permet de retrouver exactement un message appris ξ^μ . Si nous tolérons quelques erreurs dans le processus de correction, alors C. Newman ([**New**]) a démontré que la capacité de stockage devient linéaire en N dans le cas du modèle original de Hopfield. Pour cela, il a considéré la dynamique séquentielle pour laquelle on peut définir une fonction énergie (ou hamiltonien) qui décroît sur toute trajectoire induite par la dynamique. Et c'est en étudiant les propriétés de cet hamiltonien que C. Newman est parvenu à cette conclusion.

Nous allons rappeler ce résultat plus précisément avant de présenter un résultat analogue que nous avons obtenu pour le modèle de Hopfield défini sur un graphe aléatoire de Erdős-Renyi $G(N, p)$. Définissons la transformation $U : S \rightarrow S$, associée à la dynamique séquentielle déterministe. La dynamique séquentielle choisie est dite déterministe car l'ordre de mise à jour des coordonnées d'un message l'est : $U = T_N \circ \dots \circ T_1$, où les T_i ont été définies pour la dynamique du modèle de Hopfield original. Partant d'une condition initiale $\sigma(0)$, la trajectoire induite par U dans l'espace des configurations est alors définie par : $\sigma(t+1) = U(\sigma(t))$.

Par analogie avec les systèmes physiques dont les états stables sont les minima de l'énergie, nous allons définir le *hamiltonien*, ou *fonction énergie*, associée au modèle séquentiel de Hopfield, telle que ses minima correspondent aux états d'équilibre.

$$\begin{aligned} H_N : \Omega \times S &\longrightarrow \mathbf{R} \\ (\omega, \sigma) &\longmapsto H_N(\underline{\xi}(\omega), \sigma) = -\frac{1}{N} \sum_{i,j=1}^N J_{ij} \sigma_i \sigma_j \\ &= -\frac{1}{N} \sum_{\substack{i,j=1 \\ i \neq j}}^N \sum_{\mu=1}^p \xi_i^\mu(\omega) \xi_j^\mu(\omega) \sigma_i \sigma_j, \end{aligned}$$

où $\underline{\xi} = (\xi_i^\mu)_{i=1, \dots, N}^{\mu=1, \dots, M}$. Ainsi, $H_N(\underline{\xi}, \cdot)$ est une application sur S , dépendant des paramètres aléatoires $(\xi_i^\mu)_{i=1, \dots, N}^{\mu=1, \dots, M}$. Il est alors possible de démontrer quelques propriétés intéressantes (voir par exemple [3]).

PROPOSITION 10. *Pour la dynamique séquentielle déterministe de Hopfield,*

- la valeur du hamiltonien $H_N(\underline{\xi}, \cdot)$ décroît sur toute trajectoire induite par U dans l'espace des configurations S .

- un état stable est atteint pour toute condition initiale $\sigma(0)$.

- tout état stable est minimum local du hamiltonien $H_N(\underline{\xi}, \cdot)$.

REMARQUE 11. *Pour $M = 1$, il est clair que ξ^1 et $-\xi^1 = (-\xi_1^1, \dots, -\xi_N^1)$ sont les seuls minima globaux de $H_N(\underline{\xi}, \cdot)$, et $H_N(\underline{\xi}, \xi^1) = H_N(\underline{\xi}, -\xi^1) = -N$.*

En outre, dès que $M \geq 2$, la situation n'est plus du tout triviale (voir [3]).

Enonçons le théorème de Newman. Rappelons que $\mathcal{S}_{\rho N}(\xi^\mu) = \{x \in S^N, d_H(x, \xi^\mu) = \rho N\}$, où $d_H(\cdot, \cdot)$ est la distance de Hamming.

THÉORÈME 12. (Newman, 1988) [New] *Il existe un nombre $\alpha_c > 0$ tel que pour chaque $\alpha \leq \alpha_c$, $M = \alpha N$, il existe $\delta \in]0, \frac{1}{2}[$, $\epsilon > 0$, $\epsilon' \equiv f(\alpha, \delta, \epsilon) > 0$, tels que*

$$\lim_{\substack{N \rightarrow \infty \\ \frac{M}{N} \rightarrow \alpha}} e^{\epsilon' N} \left(1 - \mathbb{P} \left[\bigcap_{\mu=1}^M \bigcap_{\sigma \in \mathcal{S}_{\delta N}(\xi^\mu)} \left\{ H_N(\underline{\xi}, \sigma) > H_N(\underline{\xi}, \xi^\mu) + \epsilon N \right\} \right] \right) = 0$$

En particulier, le lemme de Borel-Cantelli implique :

$$\mathbb{P} \left[\liminf_N \left\{ \bigcap_{\mu=1}^M \bigcap_{\sigma \in \mathcal{S}_{\delta N}(\xi^\mu)} \left\{ H_N(\underline{\xi}, \sigma) > H_N(\underline{\xi}, \xi^\mu) + \epsilon N \right\} \right\} \right] = 1$$

Le théorème de Newman signifie que pour presque tout choix de messages mémorisés $\{\xi^1, \dots, \xi^M\}$, chacun des messages ξ^μ est entouré d'une barrière énergétique. Cela implique en particulier qu'il existe au moins un minimum local dans la boule de centre ξ^μ , de rayon δN . Toutefois, le minimum n'est pas nécessairement ξ^μ !

Ce théorème, avec la proposition 10, suggère que la mémoire associative avec erreurs tolérées fonctionne pour un nombre d'images mémorisées proportionnel à la taille du

réseau. Il faut également tenir compte d'un autre paramètre : $\delta(\alpha)$, représentant la fraction de bits erronés dans l'image retrouvée. Dans son article, Newman obtient les bornes rigoureuses suivantes : $\alpha_c \simeq 0,056$; et $\delta(\alpha) < 0,012$ pour $\alpha \leq 0,056$. Ces bornes ont été améliorées depuis par Loukianova ($\alpha_c \simeq 0,072$), Talagrand ($\alpha_c \simeq 0,08$) et Shcherbina *et al.* ($\alpha_c \simeq 0,11$) (voir références dans [12]). Il est intéressant de noter que Amit *et al.* ([AGS]), par des simulations numériques et des techniques de calcul non rigoureuses, ont obtenu la borne $\alpha_c \simeq 0,14$.

Dans notre article [12], Matthias Löwe et moi avons considéré ce problème dans le cas du modèle de Hopfield défini sur un graphe aléatoire de Erdős-Renyi $G(N, p)$. Le graphe $G(N, p)$ est défini de la manière suivante : soit $V = \{1, \dots, N\}$ l'ensemble des vertex du graphe. Deux sites quelconques sont alors reliés par un lien avec probabilité p : la matrice d'adjacence $A = (A_{ij})$ du graphe est telle que $A_{ij} = A_{ji}$ et les A_{ij} sont des v.a.i.d. de loi de Bernoulli, avec $P[A_{12} = 1] = 1 - P[A_{12} = 0] = p$. Remarquons que p peut dépendre de N .

Il est alors possible de définir la dynamique séquentielle déterministe $U = T_N \circ \dots \circ T_1$ pour ce modèle, avec T_i défini par l'équation (1). La fonction énergie associée à ce modèle de Hopfield dilué est alors :

$$H_N^{dil}(\sigma) = -\frac{1}{2pN} \sum_{i,j=1}^N \sum_{\mu=1}^M A_{ij} \sigma_i \sigma_j \xi_i^\mu \xi_j^\mu.$$

Soit

$$h_N(\sigma, \delta) = \min_{\sigma' \in \mathcal{S}_{\delta N}(\sigma)} H_N^{dil}(\sigma').$$

Nous dirons qu'il existe une barrière énergétique de hauteur ϵN centrée en ξ^μ s'il existe un $\delta \in]0, 1/2[$, tel que

$$h_N(\xi^\mu, \delta) \geq H_N^{dil}(\xi^\mu) + \epsilon N.$$

THÉORÈME 13 ([12]). *Soit $c_0 > 0$ et supposons que $p \geq c_0 \frac{\log N}{N}$. Alors il existe $\alpha_c > 0$ tel que si $M \leq \alpha_c pN$, il existe $\epsilon > 0$ et $\delta \in (0, 1/2)$ tel qu'il existe $\gamma > 0$ vérifiant*

$$\mathbb{P}_A[\mathbb{P}_\xi[\bigcap_{\mu=1}^M \{h_N(\xi^\mu, \delta) > H_N^{dil}(\xi^\mu) + \epsilon N\}] \geq 1 - e^{-\gamma N}] \longrightarrow 1$$

lorsque $N \longrightarrow \infty$.

Remarquons qu'il y a deux aléa ici : les (ξ_i^μ) sont aléatoires (\mathbb{P}_ξ) et le graphe est aléatoire (\mathbb{P}_A). Ce théorème montre que si p est choisi de sorte que le graphe soit connexe (avec une probabilité tendant vers 1 lorsque $N \rightarrow \infty$), alors la propriété montrée par C. Newman reste vraie, mais la capacité du modèle dilué est plus faible et dépend du paramètre p .

En 2016, D. Krotov et J. Hopfield ([KH]) ont proposé une extension du modèle de Hopfield, en remplaçant l'interaction quadratique par une dynamique plus générale. Dans l'article [18], Mete Demircigil, Judith Heusel, Matthias Löwe, Sven Uppgang et moi avons étudié la capacité de stockage de ces modèles. D. Krotov et J. Hopfield ont proposé de

remplacer la dynamique classique du modèle de Hopfield par la dynamique séquentielle déterministe $U = T_N \circ \dots \circ T_1$, où

$$T_i(\sigma) = \operatorname{sgn} \left[\sum_{\mu=1}^M (F(1 \cdot \xi_i^\mu + \sum_{j \neq i} \xi_j^\mu \sigma_j) - F((-1) \cdot \xi_i^\mu + \sum_{j \neq i} \xi_j^\mu \sigma_j)) \right] \quad (2)$$

et $F : \mathbb{R} \rightarrow \mathbb{R}$ est une fonction continue. Si on choisit $F(x) = x^2$, il est facile de voir que l'on retrouve le modèle de Hopfield original.

Si on choisit $F(x) = x^n$, pour $n \geq 2$ entier, on obtient une dynamique similaire à la dynamique induite par

$$\widehat{T}_i(\sigma) = \operatorname{sgn} \left(\sum_{1=j_1, \dots, j_{n-1}}^N \sigma_{j_1} \cdots \sigma_{j_{n-1}} W_{i, j_1 \dots j_{n-1}} \right),$$

où

$$W_{i_1, \dots, i_n} = \frac{1}{N^{n-1}} \sum_{\mu=1}^M \xi_{i_1}^\mu \xi_{i_2}^\mu \cdots \xi_{i_n}^\mu.$$

Nous avons alors le résultat suivant.

THÉORÈME 14. (1) (*Newman, 1988*) ([**New**]) *Le modèle de Hopfield généralisé associé à la dynamique $\widehat{U} = \widehat{T}_N \circ \dots \circ \widehat{T}_1$ peut stocker jusqu'à $M = \alpha_n N^{n-1}$ messages, au sens de la mémorisation avec erreurs tolérées.*

(2) ([**18**]) *Pour le même modèle, si $M = \frac{N^{n-1}}{c_n \log N}$, avec $c_n > 2(2n-3)!!$, alors pour chaque μ , la probabilité que le message ξ^μ soit un point fixe de la dynamique tend vers 1 lorsque $N \rightarrow +\infty$.*

Ce théorème montre qu'avec une interaction polynomiale d'ordre n , on obtient une capacité d'ordre N^{n-1} . Il est donc naturel de se demander s'il est possible d'augmenter encore la capacité en considérant la fonction d'interaction exponentielle $F(x) = e^x$. C'est l'objet du théorème suivant, qui montre que l'on peut atteindre une capacité exponentielle dans ce cas.

THÉORÈME 15 ([**18**]). *Considérons le modèle de Hopfield généralisé avec la dynamique définie dans (2) et la fonction d'interaction F donnée par $F(x) = e^x$. Soit $0 < \alpha < \log(2)/2$ fixé et $M = \exp(\alpha N) + 1$ supposé entier. De plus, fixons $\rho \in [0, 1/2[$. Alors pour tout μ et tout $\tilde{\xi}^\mu$ choisi au hasard dans la sphère de Hamming $\mathcal{S}_{\rho N}(\xi^\mu)$ de rayon ρN centrée en ξ^μ , où ρN est supposé être entier, nous avons*

$$\mathbb{P} \left(\exists \mu \exists i : T_i(\tilde{\xi}^\mu) \neq \xi_i^\mu \right) \rightarrow 0,$$

si α dépend de ρ et est tel que

$$\alpha < \frac{I(1-2\rho)}{2},$$

où $I : x \mapsto \frac{1}{2}((1+x) \log(1+x) + (1-x) \log(1-x))$.

Ce théorème montre qu'en changeant la dynamique du modèle de Hopfield original, on peut atteindre une capacité de stockage exponentielle. Cependant, il faut noter que le principe de ce modèle est différent du modèle original, puisqu'il requiert la connaissance de tous les messages mémorisés pour fonctionner, contrairement au modèle original, où on suppose seulement connus les poids synaptiques J_{ij} .

2. Autres modèles de mémoire associative

En 2011, ayant été membre du jury de thèse de Vincent Gripon, je me suis intéressé à des modèles de mémoire associative inspirés du modèle de Vincent Gripon et Claude Berrou ([GrB]). Vincent Gripon et Claude Berrou ont souhaité proposer des modèles biologiquement plus plausibles que le modèle de Hopfield, en introduisant notamment le concept de cliques neurales (voir [GrB]).

Dans l'article [16], Matthias Löwe, Judith Heusel et moi avons considéré un modèle similaire à leur modèle, mais muni d'une dynamique à seuil, pour lequel nous pouvons déterminer des bornes asymptotiques rigoureuses sur la capacité de stockage.

Considérons un alphabet $A = \{1, \dots, l\}$ ou bien de manière équivalente $\mathcal{A} = \{e_1, \dots, e_l\} \in \{0, 1\}^l$ et un ensemble de M messages de longueur c , choisis au hasard :

$$\mathcal{M} = \{m^\mu = (m_1^\mu, \dots, m_c^\mu) \in \mathcal{A}^c, \mu = 1, \dots, M\}.$$

Ces M messages sont codés par M cliques sur un réseau constitué de c blocs de l neurones chacun (modèle de Vincent Gripon et Claude Berrou [GrB]). Cela signifie qu'un message correspond à un graphe sur l'ensemble des $N = c l$ sites (ou neurones). Pour un message donné m^μ , le $k^{\text{ème}}$ neurone est actif dans le $i^{\text{ème}}$ bloc si $m_i^\mu = e_k$ et tous ces neurones actifs sont reliés par des liens synaptiques, ce qui définit la clique associée à ce message.

Pour ce modèle, le stockage des messages correspond à l'existence de ces cliques. Contrairement au modèle du Hopfield où les poids J_{ij} sont quantifiés, les poids synaptiques sont donc ici 1 ou 0 (un lien existe ou non).

Si trop de messages sont stockés dans le réseau, le réseau finira par être saturé et une clique associée à un message arbitraire m^0 non appris pourra figurer dans le réseau, créant la confusion avec les messages réellement appris. Le résultat suivant, dont on pourra trouver l'idée de la démonstration dans notre article [17], permet de quantifier asymptotiquement la probabilité que cela se produise, en fonction des valeurs de l, c et M .

PROPOSITION 16. *Soit m^0 un message choisi au hasard dans \mathcal{A}^c et $\mathcal{C}(m^0) = \{ \text{la clique associée au message } m^0 \text{ existe dans le réseau associé aux } M \text{ messages appris} \}$.*

Alors :

$$\mathbb{P}[\mathcal{C}(m^0)] \geq (d)^L$$

où $L = c(c-1)/2$ est le nombre de liens dans une clique et $d = 1 - (1 - 1/l^2)^M$ est la probabilité qu'un lien donné de m^0 existe.

De plus :

— Si c est fixé, $l \rightarrow \infty$ et $M = \alpha l^2$,

$$\mathbb{P}[\mathcal{C}(m^0)] = (d)^L(1 + o(1)),$$

— Si $c = \log(l)$, $l \rightarrow \infty$ et $M = \alpha l^2$, il existe une constante $C > 0$ telle que

$$(d)^L \leq \mathbb{P}[\mathcal{C}(m^0)] \leq Cc^3 M/l^3$$

En particulier, si $M = \alpha l^2$, alors $d \rightarrow 1 - e^{-\alpha}$ lorsque $l \rightarrow \infty$ et si $c = \log(l)$, alors $\mathbb{P}[\mathcal{C}(m_0)] \rightarrow 0$, lorsque $l \rightarrow \infty$.

Remarquons que nous ignorons cependant si dans le cas où $M = \alpha l^2, c = \log(l)$, nous avons $\mathbb{P}[\mathcal{C}(m^0)] = (d)^L(1 + o(1))$. Autrement dit, l'indépendance asymptotique des liens est-elle vraie ?

Nous allons maintenant définir une dynamique sur le réseau de neurones, qui permettra de retrouver les messages stockés à partir d'une connaissance partielle de ces derniers. La matrice des poids synaptiques $W \in \{0, 1\}^{cl \times cl}$ associée au réseau de cliques est définie par

$$W = \max_{\mu=1, \dots, M} (m^\mu)(m^\mu)^t$$

À cette matrice est associée la dynamique D sur $(\{0, 1\}^l)^c$: pour $v \in (\{0, 1\}^l)^c$,

$$D(v)_{(a,k)} = \bigwedge_{b=1}^c \bigvee_{r=1}^l W_{(a,k),(b,r)} v_{(b,r)},$$

qui s'écrit aussi

$$D(v)_{(a,k)} = \mathbf{1} \left\{ \sum_{b=1}^c \mathbf{1}_{\{\sum_{r=1}^l W_{(a,k),(b,r)} v_{(b,r)} \geq 1\}} \geq c \right\}$$

Définissons les variables aléatoires de Bernoulli $\zeta_{(a,j)}^\mu$

$$\zeta_{(a,i)}^\mu = \begin{cases} 1 & \text{si } m_a^\mu = e_i, \\ 0 & \text{sinon} \end{cases}$$

Nous avons $\mathbb{P}[\zeta_{(a,i)}^\mu = 1] = 1/l$. La matrice des poids synaptiques $W \in \mathbb{N}^{cl \times cl}$ est alors

$$W_{(a,i),(b,j)} = \sum_{\mu=1}^M \zeta_{(a,i)}^\mu \zeta_{(b,j)}^\mu,$$

pour $a, b \in \{1, \dots, c\}, a \neq b$ et $i, j \in \{1, \dots, l\}$. Nous posons $W_{(a,i),(a,j)} = 0$ pour tous i, j . Remarquons que les v.a. $W_{(a,i),(b,j)}$ sont de loi Binomiale $\mathcal{B}(M, 1/l^2) \sim \text{Poisson}(\alpha)$, si $M = \alpha l^2$ et l grand.

Pour $v = (v_{(b,j)})_{1 \leq b \leq c, 1 \leq j \leq l} \in \{0, 1\}^{cl}$, nous définissons l'application

$$\varphi_{(a,i)}(v) = \mathbf{1} \left\{ \sum_{b=1}^c \sum_{j=1}^l W_{(a,i),(b,j)} v_{(b,j)} \geq \kappa c \right\}$$

pour un $\kappa > 0$ fixé. Nous pouvons alors définir :

— une dynamique séquentielle $v(t+1) = S(v(t))$, où

$$S = \varphi_{(c,l)} \circ \varphi_{(c,l-1)} \circ \dots \circ \varphi_{(1,1)}$$

— la dynamique parallèle $v(t+1) = T(v(t))$, où

$$T = (\varphi_{(1,1)}, \dots, \varphi_{(l,c)})$$

Soit H_S la fonction énergie associée à S :

$$H_S(v) = -\frac{1}{2} \sum_{(a,i)} \sum_{(b,j)} W_{(a,i),(b,j)} v_{(a,i)} v_{(b,j)} + \kappa c \sum_{(a,i)} v_{(a,i)}.$$

Alors pour tout $v(0)$, $H_S(v(t+1)) \leq H_S(v(t))$ pour tout t et la dynamique séquentielle converge vers un point fixe.

Soit H_T la fonction énergie associée à T :

$$H_T(v) = - \sum_{(a,i)} \sum_{(b,j)} W_{(a,i),(b,j)} v_{(a,i)} y_{(b,j)} + \kappa c \sum_{(a,i)} (v_{(a,i)} + y_{(a,i)}),$$

où $y = T(v)$. Alors pour tout $v(0)$, $H_T(v(t+1)) \leq H_T(v(t))$ pour tout t et la dynamique parallèle converge vers un point fixe v^* , i.e. $T(v^*) = v^*$, ou un cycle limite $(\tilde{v}^1, \tilde{v}^2)$ de longueur 2, i.e. $T(\tilde{v}^1) = \tilde{v}^2$ et $T(\tilde{v}^2) = \tilde{v}^1$.

Le résultat suivant donne les conditions pour avoir la stabilité des images stockées avec grande probabilité.

THÉORÈME 17 ([16]). *Soit $c = \log l$ et $M = \alpha l^2$. Pour $\kappa \leq 1 - 1/c$, nous avons :*

(1) *Si $\alpha < \kappa$, alors pour chaque μ , chaque bloc a et chaque coordonnée k , nous avons*

$$\mathbb{P} \left(\varphi_{(a,k)}(m^\mu) = m_{(a,k)}^\mu \right) \rightarrow 1$$

lorsque l tend vers l'infini.

(2) *Si $\alpha < \kappa \exp(-(3 + \kappa)/\kappa)$, nous avons :*

$$\mathbb{P}(\forall 1 \leq \mu \leq M : S(m^\mu) = m^\mu) \rightarrow 1$$

$$\mathbb{P}(\forall 1 \leq \mu \leq M : T(m^\mu) = m^\mu) \rightarrow 1$$

lorsque l tend vers l'infini.

Pour $\kappa = 1 - 1/c$, nous pouvons ainsi montrer que pour tout $\alpha \leq \alpha^* \simeq 0.135$, $M = \alpha l^2$ messages choisis au hasard sont des points fixes de la dynamique, avec une probabilité tendant vers 1. Une question naturelle est de savoir si cette borne est optimale. Le résultat suivant y répond partiellement, en montrant que l'ordre obtenu est optimal.

THÉORÈME 18 ([16]). *Il existe un $\alpha_0 \simeq 0.45$ tel que si $c = \log(l)$ et $M = \alpha l^2$, avec $\alpha > \alpha_0$, alors*

$$\mathbb{P}(T(m^1) \neq m^1) \rightarrow 1$$

lorsque l tend vers l'infini.

Le système peut aussi corriger des erreurs aléatoires, ce qui est le principe de mémoire associative.

THÉORÈME 19 ([16]). Pour $r \in \{1, \dots, c\}$, soit

$$\mathcal{B}(m^\mu, r) = \{m \in \mathcal{A}^c : d_H(m^\mu, m) = \text{card}\{j : m_j^\mu \neq m_j\} \leq r\}$$

Soit $c = \log l$ et $M = \alpha l^2$. Soit $\gamma \in]0, 1[$, $\kappa = \min\{1 - \gamma, 1 - 1/c\}$, et $\alpha < \kappa \exp(-(1 + \kappa)/\kappa)$. Alors pour $\mu = 1, \dots, M$ et v choisi au hasard dans $\mathcal{B}(m^\mu, \gamma c - 1)$, nous avons

$$\mathbb{P}(T(v) = m^\mu) \rightarrow 1$$

lorsque l tend vers l'infini.

Ainsi, pour $\alpha > 0$ assez petit, le système peut corriger en une étape de la dynamique parallèle un message choisi au hasard ayant au plus $\gamma c - 1$ caractères erronés (ou manquants).

Une des particularités du modèle de V. Gripon et C. Berrou dans le régime $N = l c$, $c = \log l$ est le caractère *sparse* (parcimonieux ou dilué) du stockage. En effet, chaque message ne contient qu'un nombre négligeable de 1 (les autres bits valent 0) et correspond à l'activation d'un nombre négligeable de neurones ($\log l$ parmi $l \log l$). D'autres modèles de mémoire associative ayant cette caractéristique avaient été précédemment proposés par Willshaw en 1969 ([WBL-H]) et Amari en 1989 ([Am]). Dans l'article [17], Vincent Gripon, Judith Heusel, Matthias Löwe et moi avons comparé mathématiquement les performances de ces différents modèles.

Commençons par présenter le modèle d'Amari. Pour ce modèle, l'espace des configurations est $\{0, 1\}^N$ et les messages à mémoriser sont M vecteurs $\xi^\mu = (\xi_1^\mu, \dots, \xi_N^\mu)$, $\mu = 1, \dots, M$, avec $\mathbb{P}[\xi_i^\mu = 1] = p = \frac{\log N}{N}$.

Définissons le champ local

$$S_i(\sigma) = \sum_{\substack{j=1 \\ j \neq i}}^N J_{ij} \sigma_j$$

où

$$J_{ij} = \sum_{\mu=1}^M \xi_i^\mu \xi_j^\mu$$

et la dynamique associée

$$T_i(\sigma) = \Theta(S_i(\sigma) - h),$$

où $\Theta(u) = 1$ si $u \geq 0$ et 0 sinon. Ici, nous choisissons le seuil h de la forme $h = \gamma \log N$ pour un certain $\gamma > 0$.

Ceci est un choix raisonnable si nous souhaitons que (ξ^μ) soit un point fixe de la dynamique. En effet si nous considérons par exemple $\xi_i^1 = 1$, nous avons alors

$$\sum_{j \neq i} J_{ij} \xi_j^1 = \sum_{j \neq i} \xi_j^1 + \sum_{\mu \neq 1} \sum_{j \neq i} \xi_i^\mu \xi_j^\mu \xi_j^1$$

et le premier terme est d'ordre $\log N$.

Le modèle de Willshaw ([**WBL-H**]) est similiaire au modèle d'Amari, avec la restriction que les J_{ij} ne dépendent pas du nombre de messages tels que $\xi_i^\mu \xi_j^\mu = 1$. En effet, ici, nous posons

$$J_{ij} = \Theta\left(\sum_{\mu} \xi_i^\mu \xi_j^\mu - 1\right) = \begin{cases} 1 & \text{si } \exists \mu : \xi_i^\mu \xi_j^\mu = 1 \\ 0 & \text{sinon,} \end{cases}$$

pour tous $i, j \in \{1, \dots, N\}$. Pour le modèle de Willshaw, nous supposons que les ξ_i^μ sont i.i.d. à valeurs dans $\{0, 1\}$, avec $\mathbb{P}[\xi_i^\mu = 1] = p = \frac{\log N}{N}$, ou bien nous considérons M messages choisis au hasard dans l'ensemble des messages ayant exactement $c = \log N$ neurones actifs. Les deux modèles sont similaires, mais le premier est plus simple à manipuler d'un point de vue mathématique. Deux dynamiques peuvent être considérées pour ce modèle. La première est similaire à celle utilisée pour le modèle d'Amari : pour un vecteur d'entrée $\sigma \in \{0, 1\}^N$, nous posons

$$T_i(\sigma) = \Theta(\bar{S}_i(\sigma) - h)$$

avec $\bar{S}_i(\sigma) = \sum_j J_{ij} \sigma_j$ et $h = \gamma \log N$, pour un certain $\gamma > 0$. Pour le modèle de Willshaw, nous considérons \bar{S}_i au lieu de $S_i(\sigma) = \sum_{j \neq i} J_{ij} \sigma_j$, car les simulations montrent que les performances sont meilleures en prenant en compte l'auto-influence des neurones (on parle d'effet mémoire).

Dans le cas où exactement c neurones sont actifs par message, une autre dynamique est possible : pour un vecteur $\sigma \in \{0, 1\}^N$, nous calculons tous les $\bar{S}_i(\sigma)$ et les ordonnons. Notons les $h_{(1)} \geq h_{(2)} \geq \dots \geq h_{(c)} \geq \dots \geq h_{(N)}$. Alors les neurones i tels que $\bar{S}_i(\sigma) \geq h_{(c)}$ prennent la valeur 1 et les autres la valeur 0 (cette procédure est appelée algorithme « Winner Takes All », WTA).

Dans la même situation, avec c est fixé mais inconnu, nous pouvons choisir de mettre à 1 les neurones les plus actifs et mettre à 0 tous les neurones tels que $\bar{S}_i(\sigma) < h_{(1)}$. Nous verrons que si nous prenons en entrée un message partiellement effacé, cela ne change rien pour la correction des erreurs en une étape, car dans ce cas, $h_{(1)} = h_{(c)}$. Remarquons également qu'avec $h_{(1)}$, utiliser plus d'une étape de la dynamique ne change rien, alors que les performances s'améliorent si on utilise plusieurs étapes avec $h_{(c)}$ (voir les simulations dans notre article [**17**]). Remarquons également que la dynamique avec le seuil $h_{(1)}$ peut s'appliquer au cas où les messages sont composés de coordonnées ξ_i^μ i.i.d. à valeurs dans $\{0, 1\}$, avec $\mathbb{P}[\xi_i^\mu = 1] = p = \frac{\log N}{N}$.

Le modèle de Gripon et Berrou est assez similaire au modèle de Willshaw, avec la particularité d'avoir une structure par blocs : nous supposons que $N = l \log l = lc$. Les M messages ξ^1, \dots, ξ^M ont c neurones actifs, exactement un par bloc de l neurones. Pour prendre en compte la structure par blocs du réseau, nous noterons (a, k) le $k^{\text{ème}}$ neurone du $a^{\text{ème}}$ bloc.

Pour $a \neq a'$, un lien $e = ((a, k), (a', k'))$ est dit actif pour le message ξ^μ si $\xi_{(a,k)}^\mu \xi_{(a',k')}^\mu = 1$. Soit

$$\mathcal{E}((\xi^\mu)_{\mu=1, \dots, M}) = \{e : e \text{ est un lien actif d'un des } \xi^\mu\}.$$

Nous pouvons aussi définir le graphe associé à un message arbitraire ξ^0 . Il s'agit du graphe complet ayant pour sommets les (a, k) vérifiant $\xi_{(a,k)}^0 = 1$ et pour liens les $e = ((a, k), (a', k'))$, $a, a', k, k', a \neq a'$ tels que $\xi_{(a,k)}^0 \xi_{(a',k')}^0 = 1$. Alors un message ξ^0 est dit

stocké par le modèle si tous les liens du graphe complet associé à ξ^0 sont présents dans l'ensemble des liens $\mathcal{E}((\xi^\mu)_{\mu=1,\dots,M})$. Notons que si c'est le cas, ξ^0 est stocké dans le modèle au même titre que ξ^1, \dots, ξ^M .

De manière similaire au modèle de Willshaw, nous définissons l'efficacité synaptique par

$$W_{(a,k),(a',k')} = \Theta\left(\sum_{\mu=1}^M \xi_{(a,k)}^\mu \xi_{(a',k')}^\mu - 1\right).$$

Alors pour $a \neq a'$ $W_{(a,k),(a',k')} = 1$ si et seulement si (a, k) et (a', k') sont activés simultanément dans un des messages (les deux dans le même message). D'autre part, pour $a = a'$ nous avons $W_{(a,k),(a,k')} = 1$ si et seulement si $k = k'$ et il existe un μ tel que le neurone k dans le bloc a vaille 1.

Nous définissons alors la dynamique T sur $(\{0, 1\}^l)^c$: au lieu du champ local $S_i(\sigma)$ des modèles précédents, nous considérons

$$S_{(a,k)}(\sigma) = \sum_{b=1}^c \sum_{r=1}^l \Theta(W_{(a,k),(b,r)} \sigma_{(b,r)} - 1)$$

pour $\sigma \in (\{0, 1\}^l)^c$, et la dynamique

$$T_{(a,k)}(\sigma) = \Theta(S_{(a,k)}(\sigma) - h).$$

Ici aussi, h est un seuil que l'on doit choisir. Pour $h = c$, il est facile de vérifier que tous les messages appris ξ^1, \dots, ξ^M sont stables, i.e. $T(\xi) = \xi$. Cette dynamique est équivalente à la dynamique à seuil du modèle de Willshaw. Comme pour ce dernier, nous pouvons aussi définir un algorithme WTA, respectant la structure locale du modèle de Gripon et Berrou. Pour le décrire, supposons que nous souhaitons mettre à jour les valeurs des neurones du bloc a : $\sigma_{(a,k)}, k = 1, \dots, l$. Pour chaque $k = 1, \dots, l$, définissons

$$s_{(a,k)}(\sigma) = \sum_{b=1}^c \Theta\left(\sum_{r=1}^l W_{(a,k),(b,r)} \sigma_{(b,r)} - 1\right),$$

qui prend en compte que pour chaque message appris, il ne peut y avoir qu'un seul lien actif entre deux blocs. Nous ordonnons ensuite les $s_{(a,k)}, k = 1, \dots, l$ et posons les neurones ayant la plus forte valeur à 1 et les autres à 0.

Nous allons maintenant nous intéresser à la capacité de stockage de ces différents modèles. Comme nous l'avons mentionné ci-dessus, pour le modèle de Gripon et Berrou avec seuil $h = c$, les M messages appris sont stables pour la dynamique. Mais si M est choisi trop grand, des messages parasites, non appris, seront stockés par le modèle, et ne pourront être distingués des « bons » messages. Plus précisément, nous avons montré le résultat suivant, qui donne une borne supérieure sur le nombre de messages que le modèle peut mémoriser.

THÉORÈME 20 ([17]). *Considérons le modèle de Gripon et Berrou muni de la dynamique à seuil $h = c$. Soit*

$$M = \alpha(\log c)l^2 = \alpha l^2 \log \log l.$$

- Si $\alpha > 2$, un message aléatoire (indépendant des M messages appris) sera reconnu comme un message stocké avec probabilité convergant vers 1 lorsque $l \rightarrow \infty$.

- Si $\alpha = 2$ et $l \rightarrow \infty$, avec probabilité strictement positive, un message aléatoire sera considéré comme un message stocké.

- D'autre part, si $\alpha < 2$, la probabilité qu'un message aléatoire soit stocké par le modèle tend vers 0 lorsque $l \rightarrow \infty$.

Remarquons que nous avons un résultat similaire pour le modèle de Willshaw, dans le cas où les messages sont composés de coordonnées ξ_i^μ i.i.d. à valeurs dans $\{0, 1\}$, avec $\mathbb{P}[\xi_i^\mu = 1] = p = \frac{\log N}{N}$ et pour seuil $h = c$.

THÉORÈME 21 ([17]). *Considérons le modèle de Willshaw, dans le cas où les messages sont composés de coordonnées ξ_i^μ i.i.d. à valeurs dans $\{0, 1\}$, avec $\mathbb{P}[\xi_i^\mu = 1] = p = \frac{\log N}{N}$, muni de la dynamique à seuil $h = c$. Soit $M = \alpha \frac{N^2}{(\log N)^2} \log \log N$.*

- Si $\alpha > 2$, un message aléatoire (indépendant des M messages appris) sera reconnu comme un message stocké avec probabilité convergeant vers 1 lorsque $l \rightarrow \infty$.

- Si $\alpha = 2$ et $l \rightarrow \infty$, avec probabilité strictement positive, un message aléatoire sera considéré comme un message stocké.

- D'autre part, si $\alpha < 2$, la probabilité qu'un message aléatoire soit stocké par le modèle tend vers 0 lorsque $l \rightarrow \infty$.

Nous allons maintenant énoncer les résultats sur la stabilité des messages appris et la correction d'erreurs en une étape de la dynamique dans un régime où le modèle n'est pas saturé, i.e. M est supposé d'ordre $N^2/(\log N)^2$. Nous commençons avec le modèle d'Amari.

THÉORÈME 22 ([17]). *Considérons le modèle d'Amari avec seuil $h = \gamma \log N$ (avec $\gamma < 1$ à choisir), et supposons que $M = \alpha N^2/(\log N)^2$. Alors, si $\alpha < e^{-2}$, pour tout μ , nous avons*

$$\mathbb{P}(\forall i : T_i(\xi^\mu) = \xi_i^\mu) \rightarrow 1$$

lorsque $N \rightarrow \infty$.

De plus, pour chaque taux d'erreur $0 < \rho < 1$, si $\gamma < 1 - \rho$ et $\alpha < (1 - \rho)e^{-(1 + \frac{1}{1-\rho})}$, pour chaque μ , et tout message $\tilde{\xi}^\mu$ obtenu en effaçant au hasard $\rho \log N \ll 1$ dans le message ξ^μ , nous avons :

$$\mathbb{P}(\forall i : T_i(\tilde{\xi}^\mu) = \xi_i^\mu) \rightarrow 1$$

lorsque $N \rightarrow \infty$.

Enfin, si $M > -\log(1 - e^{-1})N^2/(\log N)^2$

$$\mathbb{P}(\forall i : T_i(\xi^\mu) = \xi_i^\mu) \rightarrow 0$$

lorsque $N \rightarrow \infty$.

Il est intéressant de noter que ce résultat s'applique également au modèle de Willshaw muni d'une dynamique à seuil.

COROLLAIRE 23 ([17]). *Considérons le modèle de Willshaw avec des ξ_i^μ i.i.d., un seuil $h = \gamma \log(N)$, $\gamma < 1$ et $M = \alpha N^2/(\log N)^2$, avec $\alpha < e^{-2}$. Alors pour chaque μ , nous avons*

$$\mathbb{P}(\forall i : T_i(\xi^\mu) = \xi_i^\mu) \rightarrow 1$$

lorsque $N \rightarrow \infty$.

De plus, pour chaque taux d'erreur $0 < \rho < 1$, si $\gamma < 1 - \rho$ et $\alpha < (1 - \rho)e^{-(1+\frac{1}{1+\rho})}$, pour chaque μ , et tout message $\tilde{\xi}^\mu$ obtenu en effaçant au hasard $\rho \log N \ll 1$ dans le message ξ^μ , nous avons :

$$\mathbb{P}(\forall i : T_i(\tilde{\xi}^\mu) = \xi_i^\mu) \rightarrow 1$$

lorsque $N \rightarrow \infty$.

Enfin, si $M > -\log(1 - e^{-1})N^2/(\log N)^2$

$$\mathbb{P}(\forall i : T_i(\xi^\mu) = \xi_i^\mu) \rightarrow 0$$

lorsque $N \rightarrow \infty$.

Nos simulations informatiques montrent que la dynamique utilisant la méthode WTA est plus performante que la dynamique à seuil pour le modèle de Willshaw (voir [17]). Cependant, nos résultats théoriques concernent la stabilité des messages appris et la convergence en un seul pas de la dynamique.

THÉORÈME 24 ([17]). *Considérons le modèle de Willshaw avec des messages indépendants composés de bits ξ_i^μ i.i.d. tels que $\mathbb{P}[\xi_i^\mu = 1] = \frac{c}{N}$, où $c = \log(N)$. Considérons la dynamique WTA de seuil $h_{(1)}$ et soit $M = \alpha N^2/(\log N)^2$. Alors pour $\alpha < -\log(1 - e^{-1})$, pour chaque μ , nous avons*

$$\mathbb{P}(\forall i : T_i(\xi^\mu) = \xi_i^\mu) \rightarrow 1$$

lorsque $N \rightarrow \infty$.

Cette borne est optimale : pour $\alpha > -\log(1 - e^{-1})$, pour chaque μ , nous avons

$$\mathbb{P}(\exists i : T_i(\xi^\mu) \neq \xi_i^\mu) \rightarrow 1$$

lorsque $N \rightarrow \infty$.

De plus, pour $0 \leq \rho < 1$, $\alpha < -\log(1 - e^{-1/(1-\rho)})$, pour chaque μ , et tout message $\tilde{\xi}^\mu$ obtenu en effaçant au hasard $\rho \log N \ll 1$ dans le message ξ^μ , nous avons :

$$\mathbb{P}(\forall i : T_i(\tilde{\xi}^\mu) = \xi_i^\mu) \rightarrow 1$$

lorsque $N \rightarrow \infty$.

Cette borne également est optimale : pour $\alpha > -\log(1 - e^{-1/(1-\rho)})$, nous avons pour chaque μ fixé

$$\mathbb{P}(\exists i : T_i(\tilde{\xi}^\mu) \neq \xi_i^\mu) \rightarrow 1$$

lorsque $N \rightarrow \infty$.

REMARQUE 25. Nous avons considéré dans le Théorème 24 uniquement le cas des coordonnées i.i.d.. Bien entendu, nous espérons un résultat similaire dans le cas où les messages ont exactement c neurones actifs. Mais la perte de l'indépendance rend la situation mathématiquement plus compliquée.

Nous pouvons démontrer un résultat analogue pour le modèle de Gripon et Berrou muni de la dynamique WTA.

THÉORÈME 26 ([17]). *Considérons le modèle de Gripon et Berrou muni de la dynamique WTA (notée T). Soit $M = \alpha l^2 / c^2$, avec $\alpha < -\log(1 - e^{-1})$. Alors nous avons pour chaque μ fixé*

$$\mathbb{P}(\forall(a, c) : T_{(a,c)}(\xi^\mu) = \xi_{(a,c)}^\mu) \rightarrow 1$$

lorsque $N \rightarrow \infty$.

De plus, pour $0 \leq \rho < 1$, $\alpha < -\log(1 - e^{-1/(1-\rho)})$, pour chaque μ , et tout message $\tilde{\xi}^\mu$ obtenu en effaçant au hasard $\rho \log N \ll 1$ dans le message ξ^μ , nous avons :

$$\mathbb{P}(\forall(a, c) : T_{(a,c)}(\tilde{\xi}^\mu) = \xi_{(a,c)}^\mu) \rightarrow 1$$

lorsque $N \rightarrow \infty$.

Nous avons également comparé les différents modèles par simulation (voir [17]).

Deuxième partie

Liens entre le modèle de Hopfield et le système CDMA

En lisant des articles de Remco Van der Hofstad (Université de Eindhoven), sur le modèle CDMA, et en discutant avec lui, Matthias Löwe et moi avons réalisé qu'il y a une forte similarité entre ce modèle et le modèle de Hopfield. Nous avons alors étudié ce modèle, très utilisé en télécommunications, et écrit les deux articles [8] et [10], que je vais présenter dans cette partie.

3. Le modèle CDMA

Le modèle CDMA (*Code Division Multiple Access*) est utilisé dans le système de télécommunications 3G pour permettre la transmission simultanée de messages de différents utilisateurs. Une suite de codage est attribuée à chaque utilisateur, qui permet d'encoder les différents messages avant transmission et le récepteur, qui connaît ces suites, peut ainsi retrouver les différents messages originaux. Bien entendu, ce système a ses limites : si le nombre de messages ainsi regroupés est trop important, les interférences engendrent des erreurs lors du décodage. Nous allons nous intéresser à l'étude mathématique de la capacité de ce système : quel est le nombre maximal d'utilisateurs d'un tel système ?

Définissons plus précisément le modèle CDMA : supposons que k utilisateurs souhaitent transmettre leurs signaux. Le message du $m^{\text{ème}}$ utilisateur est donné par $b_m(t)$, où $b_m(t) = b_{m\lfloor t/T \rfloor} \in \{-1, +1\}$ pour $1 \leq m \leq k$ et

$$b_m = (\dots, b_{m,-1}, b_{m0}, b_{m1}, \dots) \in \{-1, +1\}^{\mathbb{Z}}.$$

Ici pour tout nombre réel x , $\lceil x \rceil$ désigne le plus petit entier supérieur ou égal à x .

Pour coder le signal, pour chaque m , $1 \leq m \leq k$, nous avons une suite

$$a_m = (\dots, a_{m,-1}, a_{m0}, a_{m1}, \dots) \in \{-1, +1\}^{\mathbb{Z}}.$$

Nous posons $a_m(t) = a_{m\lfloor t/T_c \rfloor}$, où $T_c = T/n$ et n , appelé gain de traitement, est la longueur de la séquence d'étalement de chaque utilisateur (en pratique, n est compris entre 32 et 512). Le signal codé du $m^{\text{ème}}$ utilisateur est alors

$$s_m(t) = \sqrt{2P_m} b_m(t) a_m(t) \cos(\omega_c t), \quad 1 \leq m \leq k,$$

où P_m est la puissance du $m^{\text{ème}}$ utilisateur et ω_c la fréquence porteuse. Le facteur $\cos(\omega_c t)$ est utilisé pour transmettre le signal à la fréquence ω_c . Le code $a_m(t)$ est connu par le téléphone portable du transmetteur et par la station de base.

Le signal total transmis est alors

$$r(t) = \sum_{j=1}^k s_j(t).$$

Pour simplifier, nous supposons que tous les utilisateurs utilisent la même grille temporelle et que le canal est sans bruit (le message reçu coïncide avec le message transmis).

Pour retrouver le bit b_{m1} dans le message $r(t)$, ce dernier est multiplié par $a_m(t) \cos(\omega_c t)$ et intégré sur $[0, T]$. Par simplicité, nous choisissons $\omega_c T_c = \pi f_c$, où $f_c \in \mathbb{N}$. Ainsi,

$$\frac{1}{T} \int_0^T r(t) a_m(t) \cos(\omega_c t) dt = \frac{1}{2} \sqrt{2P_m} b_{m1} + \sum_{\substack{j=1 \\ j \neq m}}^k \frac{1}{2} \sqrt{2P_j} b_{j1} \frac{1}{n} \sum_{i=1}^n a_{ji} a_{mi} \quad (3)$$

Cette procédure est souvent appelé « filtre adapté » (*Matched Filter* : MF). L'équation (3) montre que le signal ainsi décodé est la somme du bit cherché (multiplié par $\frac{1}{2}\sqrt{2P_m}$) et de l'interférence due aux autres utilisateurs. Si les vecteurs (a_{m1}, \dots, a_{mn}) et (a_{j1}, \dots, a_{jn}) , $j = 1 \dots, k, j \neq m$, étaient orthogonaux, tels que

$$\sum_{i=1}^n a_{ji}a_{mi} = 0,$$

alors les termes d'interférence seraient tous nuls. Cependant, en pratique, les suites sont souvent générées par un générateur pseudo-aléatoire et sont seulement pseudo-orthogonales. Nous les modélisons par un tableau (A_i^m) , $m = 1, 2, \dots, k, i = 1, 2, \dots, n$ de variables aléatoires i.i.d. à valeurs dans $\{-1, +1\}$, telles que

$$\mathbb{P}(A_i^m = +1) = \mathbb{P}(A_i^m = -1) = \frac{1}{2}.$$

Dans ce cas, le signal (3) s'écrit

$$Z_m^{(1)} = \frac{1}{2}\sqrt{2P_m} b_{m1} + \sum_{\substack{j=1 \\ j \neq m}}^k \frac{1}{2}\sqrt{2P_j} b_{j1} \frac{1}{n} \sum_{i=1}^n A_i^j A_i^m,$$

et le bit b_{m1} peut être estimée par

$$\hat{b}_{m1}^{(1)} = \text{sgn}(Z_m^{(1)}),$$

où $\text{sgn}(\cdot)$ est la fonction signe, et $\text{sgn}(0) = U$ pour U variable aléatoire telle que

$$\mathbb{P}(U = +1) = \mathbb{P}(U = -1) = \frac{1}{2}$$

(à chaque fois qu'on doit choisir $\text{sgn}(0)$, on prend une nouvelle réalisation de la v.a. U). Cette procédure de *Matched Filter* est utilisée dans le système de télécommunications 3G. L'indice ⁽¹⁾ indique que potentiellement, $\hat{b}_{m1}^{(1)}$ est seulement une première étape et nous verrons qu'il est possible d'améliorer les performances en itérant des étapes de décodage.

Nous allons dans un premier temps analyser les performances de ce premier décodage et estimer les probabilités d'avoir une erreur, i.e.

$$\mathbb{P}(\hat{b}_{m1}^{(1)} \neq b_{m1}) \quad \text{et} \quad \mathbb{P}(\exists m : \hat{b}_{m1}^{(1)} \neq b_{m1})$$

et étudier pour quelles valeurs de k (dépendant de n), cette probabilité tend vers 0, lorsque n tend vers l'infini. Dans l'article [8], nous avons montré le résultat suivant, qui donne la borne optimale pour k , pour un décodage sans erreur lorsque k et n tendent vers l'infini :

THÉORÈME 27 ([8]). *Supposons que $k = \frac{n}{\gamma \log n}$ pour un certain $\gamma > 0$ et que toutes les puissances P_m sont égales. Alors, si $\gamma > 2$,*

$$\lim_{n \rightarrow \infty} \mathbb{P}(\hat{b}_m^{(1)} = b_m, \forall m = 1, \dots, k) = 1,$$

tandis que si $\gamma < 2$,

$$\lim_{n \rightarrow \infty} \mathbb{P}(\hat{b}_m^{(1)} = b_m, \forall m = 1, \dots, k) = 0.$$

Nous avons vu que les erreurs dans la méthode par filtre adapté proviennent des termes d'interférence avec les autres utilisateurs dans l'équation (3) (i.e. les termes d'indices $j \neq m$). Nous allons maintenant présenter deux méthodes permettant d'améliorer les performances de cette première étape. L'idée est d'éliminer au mieux ces termes d'interférence, d'où les noms en anglais : *soft decision parallel interference cancellation* (SD-PIC) et *hard decision parallel interference cancellation* (HD-PIC). A partir de notre première estimation $\hat{b}_{m,1}^{(1)}$ pour $b_{m,1}$, nous obtenons une estimation du signal $s_m(t)$ envoyé par le $m^{\text{ème}}$ utilisateur : pour $t \in [0, T]$,

$$\hat{s}_m^{(1)}(t) = h_m(Z_m^{(1)}) a_m(t) \cos(\omega_c t), \quad 1 \leq m \leq k,$$

où

$$h_m(x) = \begin{cases} \sqrt{2P_m} \operatorname{sgn}(x) & \text{pour la méthode HD-PIC,} \\ 2x & \text{pour la méthode SD-PIC.} \end{cases}$$

Evidemment, pour HD-PIC, nous devons connaître les puissances P_m , ce qui n'est pas le cas pour SD-PIC. L'interférence totale subie par le $m^{\text{ème}}$ utilisateur peut alors être estimée par

$$\hat{r}_m^{(1)}(t) = \sum_{j \neq m} \hat{s}_j^{(1)}(t).$$

En l'absence de bruit lors de la transmission, notons que c'est la seule source possible d'erreurs. Nous pouvons alors améliorer notre estimation de $b_{m,1}$ en soustrayant l'interférence totale du signal. Nous obtenons alors

$$\hat{b}_{m,1}^{(2)} = \operatorname{sgn}(Z_m^{(2)}),$$

où $Z_m^{(2)}$ est obtenu en remplaçant $r(t)$ dans l'équation (3) par $r(t) - \hat{r}_m^{(1)}(t)$, de façon que, si toutes les puissances sont supposées égales à 2 pour simplifier,

$$Z_m^{(2)} = b_{m,1} + \sum_{\substack{j=1 \\ j \neq m}}^k \frac{1}{n} \sum_{i=1}^n A_i^j A_i^m (b_{j,1} - h_j(Z_j^{(1)})).$$

Cette procédure peut être itérée. Après $s - 1$ pas de suppression d'interférences (remarquons en effet qu'il n'y a pas de suppression d'interférences à la première étape), nous obtenons $\hat{b}_{m,1}^{(s)} = \operatorname{sgn}(Z_m^{(s)})$, où

$$Z_m^{(s)} = b_{m,1} + \sum_{\substack{j=1 \\ j \neq m}}^k \frac{1}{n} \sum_{i=1}^n A_i^j A_i^m (b_{j,1} - h_j(Z_j^{(s-1)})).$$

Nous allons maintenant étudier comment se comporte $\mathbb{P}(\exists m : \hat{b}_{m,1}^{(s)} \neq b_{m,1})$. En particulier, peut-on ainsi augmenter le nombre d'utilisateurs si nous autorisons plusieurs étapes de suppression d'interférences ?

Dans notre article [8], nous obtenons un premier résultat lorsque s est fixé (i.e. ne dépend pas de n) :

THÉORÈME 28 ([8]). *Supposons que $k = \frac{n}{\gamma \log n}$ et que toutes les puissances P_m sont égales. Pour tout $s \geq 2$ et $\gamma > \frac{2}{s}$,*

$$-\limsup_{n \rightarrow \infty} \frac{\sqrt[s]{k}}{n} \log \mathbb{P}(\exists m = 1, \dots, k : \hat{b}_m^{(s)} \neq b_m) > 0.$$

Appliquer $s - 1$ étapes de la méthode HD-PIC permet donc d'augmenter le nombre d'utilisateurs par s , par rapport à la méthode par filtre adaptée. La démonstration du théorème 28 est une extension de résultats montrés dans les articles [HK1] et [HK2]. Dans l'article [FHK], un résultat similaire est démontré pour le SD-PIC, lorsque $k = o(\frac{n^{\frac{s+1}{s}}}{\log n})$.

En s'appuyant sur les résultats [FHK] et [HK1], nous obtenons un nombre d'utilisateurs possible d'ordre n , si le nombre d'étapes peut augmenter avec n :

THÉORÈME 29 ([8]). *Supposons que $k = \delta n$ pour un certain $\delta > 0$ et que toutes les puissances P_m sont égales.*

(i) *Pour SD-PIC et $\delta < (\sqrt{2} - 1)^2$, il existe un $M = M(\delta)$ tel que pour tout $s \geq M \log n$,*

$$\lim_{n \rightarrow \infty} \mathbb{P}(\hat{b}_m^{(s)} \neq b_m \forall m = 1, \dots, k) = 1. \quad (4)$$

(ii) *Pour HD-PIC et $\delta < \frac{\frac{1}{2} \log 2 - \frac{1}{4}}{2 \log 2}$, il existe un $M = M(\delta)$ tel que pour tout $s \geq M \log n$,*

$$-\limsup_{n \rightarrow \infty} \frac{1}{n} \log \mathbb{P}(\exists m = 1, \dots, k : \hat{b}_m^{(s)} \neq b_m) > 0. \quad (5)$$

En fait, nos simulations (voir [8]) montrent qu'appliquer une seule étape ($s = 2$) de la suppression d'interférences permet déjà d'améliorer considérablement les performances du système. Nous avons étudié plus précisément ce cas particulier dans le cas SD-PIC et obtenu le résultat suivant :

THÉORÈME 30 ([10]). *Supposons que $k = \frac{n}{\sqrt{\gamma \log n}}$, avec $\gamma > 2$ et que toutes les puissances P_m sont égales. Alors*

$$\lim_{n \rightarrow \infty} \mathbb{P}(\hat{b}_{m,1}^{(2)} = b_{m,1}, \forall m = 1, \dots, k_n) = 1.$$

REMARQUE 31. *Les simulations montrent que le HD-PIC est plus performant que le SD-PIC, mais nous n'avons pas pu montrer un résultat similaire pour le HD-PIC. En effet, la fonction sgn dans la dynamique rend la situation beaucoup plus complexe que le cas SD-PIC, où la fonction h_m est linéaire.*

4. Application au modèle de Hopfield

Nous nous sommes rendus compte avec Matthias Löwe que la méthode SD-PIC pour le modèle CDMA est similaire à une dynamique proposée par Amari et Yanai ([AY]) en 1996 pour améliorer les capacités de stockage du modèle de Hopfield : ils conjecturent que leur dynamique leur permet de passer d'une capacité d'ordre $N/\log(N)$ à une capacité d'ordre $N/\sqrt{\log N}$ pour le modèle de Hopfield. Remarquons que ceci est similaire à l'application d'une étape du SD-PIC pour le modèle CDMA (Théorème 30), qui permet d'améliorer

la méthode par filtre adapté. Nous avons prouvé ce résultat, publié mais non démontré rigoureusement dans l'article [AY].

La dynamique proposée par Amari et Yanai peut être vue comme une modification de la dynamique séquentielle utilisée pour le modèle de Hopfield classique et est liée à la méthode d'apprentissage par pseudo-inverse (voir [10], Remarque 3.2). Nous considérons le formalisme usuel du modèle de Hopfield (voir section 1 de la première partie) et définissons la dynamique séquentielle $T = T_N \circ \dots \circ T_1 : \{-1, +1\}^N \rightarrow \{-1, +1\}^N$, où

$$T_i(\sigma) = \operatorname{sgn} \left(\sum_{j=1}^N (J - aJ^2)_{ij} \sigma_j \right), \quad i = 1, \dots, N.$$

Ici $a \in (0, 1)$ est un paramètre et $J = (J_{ij})_{\substack{i=1, \dots, N \\ j=1, \dots, N}}$ est la matrice des poids synaptiques J_{ij} définis par la règle de Hebb :

$$J_{ij} = \frac{1}{N} \sum_{\mu=1}^M \xi_i^\mu \xi_j^\mu, \quad \text{pour } i, j \in \{1, \dots, N\}.$$

Contrairement à nous, Amari et Yanai posent $J_{ii} = 0$ pour $i = 1, \dots, N$, mais cette différence est mineure pour N grand et ne modifie pas la capacité du modèle. Nous avons alors le résultat suivant à propos de la stabilité des messages appris :

THÉORÈME 32 ([10]). *Sous les hypothèses que les v.a. $(\xi_i^\mu)_{i,\mu}$ sont i.i.d. avec*

$$\mathbb{P}(\xi_i^\mu = +1) = \mathbb{P}(\xi_i^\mu = -1) = \frac{1}{2},$$

nous avons :

(i) *Si $a = \frac{1}{2}$ et M est tel que $M = c \frac{N}{\sqrt{2 \log N}}$, pour un certain $c \in (0, 1)$, alors pour tout $\mu = 1, \dots, M$,*

$$\lim_{N \rightarrow +\infty} \mathbb{P}[T(\xi^\mu) = \xi^\mu] = 1.$$

(ii) *Si $a \in]0, 1[\setminus \{\frac{1}{2}\}$ et M est tel que $M = c \frac{N}{\log N}$, pour un certain $c \in (0, \frac{(1-a)^2}{2(1-2a)^2})$, alors pour tout $\mu = 1, \dots, M$,*

$$\lim_{N \rightarrow +\infty} \mathbb{P}[T(\xi^\mu) = \xi^\mu] = 1.$$

REMARQUE 33. *Remarquons que pour $a \in]0, 1[\setminus \{\frac{1}{2}\}$, la capacité est du même ordre $cN/\log N$ que pour le modèle de Hopfield classique ($a = 0$), avec une constante plus grande lorsque $a \in]0, 2/3[$.*

5. Un résultat de déviations modérées

Nous terminons cette partie par un résultat de déviations modérées pour des sommes convenablement normalisées de variables aléatoires de Bernoulli faiblement corrélées. Il se trouve que nous avons utilisé ce résultat pour démontrer les théorèmes 30 et 32.

Considérons un tableau de v.a. i.i.d. $(A_i^j), i = 1, \dots, n, j = 1, \dots, k$ avec

$$\mathbb{P}(A_i^j = 1) = \mathbb{P}(A_i^j = -1) = \frac{1}{2} \quad \text{pour tous } i, j.$$

La quantité qui nous intéresse est

$$\Xi_n = \sum_{i_1=1}^n \sum_{\substack{i_2=1 \\ i_2 \neq i_1}}^n \sum_{j_1=1}^k \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^k A_{i_1}^{j_1} A_{i_2}^{j_1} A_{i_2}^{j_2}.$$

Bien évidemment, $\mathbb{E}[\Xi_n] = 0$ pour tout n . Nous analysons les déviations modérées de Ξ_n à l'échelle $\frac{1}{n^2}$, i.e. nous considérons

$$X_n = \frac{1}{n^2} \Xi_n.$$

Le choix de cette normalisation est liée aux applications des sections précédentes et il est sûrement possible de considérer des normalisations différentes, pour lesquelles nos techniques fonctionnent.

Notons que la bonne normalisation pour que la loi des grands nombres et donc un principe de grande déviations soient valides est $\frac{1}{n^2 k^2}$, tandis que le Théorème Central Limite est vrai si on normalise avec $\frac{1}{nk}$ (voir [De]). Nous sommes donc dans un régime de déviations modérées lorsque $k = o(n)$.

THÉORÈME 34 ([10]). *Pour X_n défini ci-dessus et $k \geq n^\beta$, $\frac{17}{18} < \beta < 1$, avec $k = o(n)$, et tout $\gamma > 0$, nous avons*

$$\lim_{n \rightarrow \infty} \frac{k^2}{n^2} \log \mathbb{P}[X_n \geq \gamma] = -\frac{\gamma^2}{2}$$

En particulier, pour n assez grand,

$$\mathbb{P}[X_n \geq \gamma] \leq 2 e^{-\frac{\gamma^2}{2} \frac{n^2}{k^2}}$$

REMARQUE 35. *Pour k trop petit, les estimations précédentes ne peuvent être vraies : R. Van der Hofstad et M. Klok ont montré par exemple dans [HK2] que pour $k = 2$, la vitesse de convergence est d'ordre e^{-cN} et non e^{-cN^2} . Il serait intéressant de connaître la valeur optimale de β telle que le théorème 34 et le lemme ci-dessous soient vrais. A ma connaissance, cette question est ouverte.*

La démonstration de ce théorème repose sur le lemme suivant que nous avons également démontré.

LEMME 36 ([10]). *Soit $p = \mathcal{O}(\frac{n^2}{k^2})$ un entier pair et $k \geq n^\beta$, $\beta > \frac{17}{18}$, avec $k = o(n)$, nous avons*

$$\mathbb{E}\left[\left(\frac{1}{n^2} \sum_{i_1=1}^n \sum_{\substack{i_2=1 \\ i_2 \neq i_1}}^n \sum_{j_1=1}^k \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^k A_{i_1}^{j_1} A_{i_2}^{j_1} A_{i_2}^{j_2}\right)^p\right] = \frac{p!}{2^{p/2} (p/2)! n^p} (1 + \varepsilon_{n,p}), \quad (6)$$

où $\lim_{n \rightarrow \infty} \varepsilon_{n,p} = 0$.

Troisième partie

Etude d'un algorithme stochastique : la méthode du swapping

Cette partie correspond aux articles [11] et [14] des travaux joints.

Les algorithmes MCMC (*Markov Chain Monte Carlo*) sont très utilisés dans de nombreux domaines pour simuler des échantillons de lois de probabilité complexes qui ne peuvent être simulés par une méthode élémentaire. Parmi ces algorithmes, l'algorithme de Metropolis-Hastings est un des plus connus, mais lorsque la distribution à simuler est multimodale, la convergence de cet algorithme peut être très lente. Ce phénomène, appelé métastabilité en physique statistique, a également été observé pour la dynamique de Glauber.

Plusieurs modifications de l'algorithme de Metropolis ont été proposées pour accélérer la convergence : parmi ces algorithmes, citons la méthode de *simulated tempering* proposée par Marinari et Parisi en 1992 ([MP]) et la méthode du *swapping* proposée par Geyer en 1995 ([Ge]). Bien que ces algorithmes aient été alors beaucoup utilisés, notamment pour des modèles issus de la physique statistique, peu de résultats théoriques justifiaient leur intérêt par rapport à l'algorithme de Metropolis. Un des premiers résultats en ce sens a été démontré par Madras et Zheng en 2003 pour le modèle de Curie Weiss ([MaZ]). En effet, ils ont montré que l'algorithme de *swapping* a un temps de convergence polynomial à toute température pour le modèle de Curie Weiss, alors qu'il est bien connu que le temps de convergence à basse température est exponentiel pour l'algorithme de Metropolis (voir par exemple [MaP]). D'un autre côté, Bhatnagar et Randall ont montré en 2004 que les deux algorithmes de *swapping* et *simulated tempering* convergent lentement pour le modèle de Potts à 3 états ([BhR]) à basse température et conjecturent que cela est dû à la transition de phase du premier ordre pour ce modèle (le modèle de Curie Weiss admet quant à lui une transition de phase du second ordre).

Ces deux résultats nous ont conduits, Matthias Löwe et moi, à nous intéresser à ces algorithmes. En adaptant les techniques utilisées par Madras et Zheng, nous avons montré en 2009 que l'algorithme de *swapping* converge à une vitesse polynomiale pour un exemple de modèle désordonné : le modèle de Hopfield stockant deux messages ([11]). En 2009, Ebbers et Löwe ont montré que la conjecture de Bhatnagar et Randall n'est pas vraie : ils ont prouvé que l'algorithme de *swapping* converge lentement pour le modèle REM (*Random Energy Model*), bien qu'il admette une transition de phase du troisième ordre ([EL]). Mirko Ebbers, Matthias Löwe et moi avons alors analysé la conjecture de Bhatnagar et Randall sur un autre modèle ordonné, le modèle de Blume-Emery-Griffiths (BEG), qui est très intéressant car il présente deux transitions de phase de natures différentes. Nous avons obtenu des résultats cohérents avec cette conjecture, en mettant en évidence pour le même modèle une vitesse de convergence respectivement exponentielle ou polynomiale selon la nature de la transition de phase ([14]).

Nous allons présenter dans la première section les algorithmes mentionnés ci-dessus, puis, dans les sections suivantes, les résultats obtenus pour les modèles de Hopfield et de Blume-Emery-Griffiths.

6. Les algorithmes de *simulated tempering* et de *swapping*

Considérons que la loi de probabilité dont nous souhaitons simuler une réalisation est une mesure de Gibbs sur un ensemble fini Ω . Pour être plus précis, soit $H(\cdot)$ une fonction énergie (ou Hamiltonien) définie sur Ω . Pour chaque $\beta > 0$, inverse de la température en

physique statistique, la mesure de Gibbs associée à H est alors

$$\pi_\beta(\sigma) = \frac{e^{-\beta H(\sigma)}}{\sum_{\sigma' \in \Omega} e^{-\beta H(\sigma')}} = \frac{e^{-\beta H(\sigma)}}{Z(\beta)}$$

Soit \mathcal{K}_{gen} une chaîne de Markov apériodique, symétrique et irréductible sur Ω . La chaîne de Markov de Metropolis-Hastings $T_\beta(\cdot, \cdot)$ est alors définie par

$$T_\beta(x, y) = \begin{cases} \mathcal{K}_{gen}(x, y) & \text{si } x \neq y \text{ et } \pi_\beta(y) \geq \pi_\beta(x) \\ \mathcal{K}_{gen}(x, y) \frac{\pi_\beta(y)}{\pi_\beta(x)} & \text{si } x \neq y \text{ et } \pi_\beta(y) < \pi_\beta(x) \\ 1 - \sum_{z \neq x} T_\beta(x, z) & \text{sinon.} \end{cases}$$

Pour les mesures de Gibbs sur un ensemble fini ayant une structure de voisinage, on choisit généralement pour \mathcal{K}_{gen} une marche aléatoire aux plus proches voisins. Par exemple, si $\Omega = \{-1, +1\}^N$, avec N entier, on peut choisir

$$K_{gen}(x, y) = \begin{cases} \frac{1}{N} & \text{si } d_H(x, y) = 1 \\ 0 & \text{sinon.} \end{cases}$$

Ici $d_H(x, y)$ est la distance de Hamming, i.e. le nombre de coordonnées où x and y diffèrent.

Il est facile de vérifier que T_β est réversible pour π_β et converge donc vers π_β par le théorème de convergence des chaînes de Markov. Cet algorithme est simple à mettre en oeuvre, mais comme nous l'avons déjà mentionné, il peut avoir un temps de convergence exponentiel, notamment pour des mesures de Gibbs à basse température (voir[MaP] pour le modèle de Curie Weiss).

6.1. L'algorithme de simulated tempering. Pour accélérer la convergence, considérons $\Omega \times \{0, 1, \dots, M\}$, avec $M \in \mathbb{N}$. Dans le cas de mesures de Gibbs sur un ensemble $\Omega = \mathcal{S}^N$, où \mathcal{S} est un ensemble fini contenant plus d'un élément et N est un entier, nous choisirons typiquement M de la forme $M = c_1 N$, où c_1 est une constante strictement positive. La seconde composante du nouvel espace des états $\Omega \times \{0, 1, \dots, M\}$ fait référence à l'inverse de la température du modèle. Définissons

$$\beta_i = \frac{i}{M} \beta \text{ et la mesure de probabilité } \pi_i = \pi_{\beta_i}.$$

$$\pi(x) = \pi((\sigma, i)) = \frac{1}{M+1} \pi_i(\sigma).$$

Nous construisons une chaîne de Markov pouvant réaliser une transition de $(\sigma, i) \in \Omega \times \{0, 1, \dots, M\}$ vers un nouvel état (σ', i) selon T_{β_i} . Pour $i \in \{0, \dots, M\}$ fixé, notons $P_{st}((\sigma, i), (\sigma', i)) = T_{\beta_i}(\sigma, \sigma')$. La température peut également être modifiée selon une chaîne de Metropolis. L'idée est que si le système est dans un puits d'énergie, il peut accroître la température (i.e. diminuer β) afin de faciliter les transitions vers des configurations hors du puits. Plus précisément : pour $\sigma \in \Omega$ fixé, une transition de (σ, i) vers (σ, j) a lieu selon les probabilités de transition

$$Q((\sigma, i), (\sigma, j)) = \begin{cases} K_{tm}(i, j) & \text{si } \pi_j(\sigma) \geq \pi_i(\sigma) \text{ et } i \neq j \\ K_{tm}(i, j) \frac{\pi_j(\sigma)}{\pi_i(\sigma)} & \text{si } \pi_j(\sigma) < \pi_i(\sigma) \\ 1 - \sum_{k \neq i} Q((\sigma, i), (\sigma, k)) & \text{si } i = j \end{cases}$$

où

$$K_{\text{tm}}(i, j) = \begin{cases} \frac{1}{2(M+1)} & \text{si } j = i \pm 1 \text{ and } j \in \{0, \dots, M\} \\ 0 & \text{si } |i - j| > 1 \\ 1 - \sum_{k \neq i} K_{\text{tm}}(i, k) & \text{si } i = j. \end{cases}$$

L'algorithme *simulated tempering* consiste en une étape de transition de température Q , une étape de la transition permettant de modifier σ (selon la chaîne de Metropolis P_{st} à la température courante) et finalement une autre étape de transition de la température. Ainsi, en termes de matrices de transition, la matrice de transition de la chaîne associée à l'algorithme *simulated tempering* est donnée par $QP_{st}Q$.

L'inconvénient de cet algorithme est qu'il requiert le calcul de $\frac{\pi_j(\sigma)}{\pi_i(\sigma)}$ dans la matrice Q et donc la connaissance des constantes de normalisation $Z(\beta_i)$ and $Z(\beta_j)$, qui sont difficiles à obtenir en général. C'est la raison pour laquelle nous introduisons un autre algorithme, l'algorithme du *swapping*.

6.2. L'algorithme de swapping. Dans l'algorithme de *swapping* proposé par Geyer ([Ge]), l'idée de pouvoir modifier la température est gardée. Comme espace des états, nous définissons

$$\Omega^{\text{sw}} = \Omega^{M+1}$$

et comme mesure de probabilité sur Ω^{sw}

$$\pi(x) = \prod_{i=0}^M \pi_i(x_i) = \frac{\prod_{i=0}^M e^{\frac{i\beta}{M} H(x_i)}}{\prod_{i=0}^M Z(\beta_i)},$$

avec $x = (x_0, \dots, x_M) \in \Omega^{\text{sw}}$. Comme pour l'algorithme de *simulated tempering*, l'algorithme de *swapping* consiste en deux étapes. Une étape consiste à choisir $i \in \{0, \dots, M\}$ uniformément et à mettre à jour la $i^{\text{ème}}$ coordonnée de l'état courant $x = (x_0, \dots, x_M)$ selon la chaîne de Metropolis T_{β_i} : la probabilité de transition de $x = (x_0, \dots, x_{i-1}, x_i, x_{i+1}, \dots, x_M) \in \Omega^{\text{sw}}$ vers $x' = (x_0, \dots, x_{i-1}, x'_i, x_{i+1}, \dots, x_M)$ est $T_i(x, x') = T_{\beta_i}(x_i, x'_i)$.

Pour tous u, v , soit $\delta(u, v) = 1$ si $u = v$ et 0 sinon. Alors la chaîne produit

$$P(x, y) = \frac{1}{2} \delta(x, y) + \frac{1}{2(M+1)} \sum_{i=0}^M \delta(x_0, y_0) \cdots \delta(x_{i-1}, y_{i-1}) T_i(x_i, y_i) \\ \times \delta(x_{i+1}, y_{i+1}) \cdots \delta(x_M, y_M) \quad (7)$$

nous donne une chaîne de Markov sur Ω^{sw} .

Dans la deuxième étape, nous choisissons $i \in \{0, \dots, M-1\}$ selon la loi uniforme et échangeons les coordonnées x_i and x_{i+1} de x avec probabilité

$$\min \left(1, \frac{\pi(x_0, \dots, x_{i+1}, x_i, \dots, x_M)}{\pi(x_0, \dots, x_i, x_{i+1}, \dots, x_M)} \right).$$

Les probabilités de transition de $x = (x_0, \dots, x_i, x_{i+1}, \dots, x_M)$ à $x' = (x_0, \dots, x_{i+1}, x_i, \dots, x_M)$ sont alors

$$Q(x, x') = \begin{cases} K_{\text{sw}}(x, x') & \text{si } \pi(x') \geq \pi(x) \text{ et } x \neq x' \\ K_{\text{sw}}(x, x') \frac{\pi(x')}{\pi(x)} & \text{si } \pi(x') < \pi(x) \\ 1 - \sum_{z \neq x} Q(x, z) & \text{si } x = x'. \end{cases}$$

K_{sw} est défini par

$$K_{\text{sw}}(x, x') = \begin{cases} \frac{1}{2M} & \text{si } \exists i \text{ tel que } x_j = x'_j \ \forall j \notin \{i, i+1\}, \\ & \text{et } x_i = x'_{i+1}, x_{i+1} = x'_i \\ 0 & \text{si } \nexists i \text{ tel que } x_j = x'_j \ \forall j \notin \{i, i+1\}, \\ & \text{et } x_i = x'_{i+1}, x_{i+1} = x'_i \\ 1 - \sum_{z \neq x} K_{\text{sw}}(x, z) & \text{si } x = x' \end{cases}$$

Le facteur $\frac{1}{2}$ dans la définition de K_{sw} et P garantit que P et Q sont apériodiques et que les opérateurs correspondants sont positifs. Remarquons que les constantes de normalisation dans Q and P disparaissent, si bien que les probabilités de transition peuvent être facilement calculées.

L'algorithme de *swapping* est une combinaison de P et Q et usuellement on prend QPQ , qui est réversible pour π si Q et P le sont (ce qui est le cas dans notre situation).

Le théorème suivant établit un lien entre les vitesses de convergence des deux algorithmes précédents.

THÉORÈME 37 (Zheng [Zh]). *S'il existe une constante $\delta > 0$ telle que*

$$\sum_{x \in \Omega} \min\{\pi_i(x), \pi_{i+1}(x)\} \geq \delta \quad \text{pour tout } 1 \leq i \leq M$$

et si l'algorithme de swapping converge en un temps polynomial, alors il en est de même pour l'algorithme de simulated tempering.

7. Application au modèle de Hopfield

Considérons le modèle de Hopfield classique, stockant p messages $\xi^\mu = (\xi_i^\mu) \in \{-1, +1\}^N$, $\mu = 1, \dots, p$, de poids synaptiques

$$J_{i,j} = \frac{1}{N} \sum_{\mu=1}^p \xi_i^\mu \xi_j^\mu, \quad \text{pour } i, j \in 1, \dots, N.$$

La fonction énergie de ce modèle est

$$H_N(\sigma) = -\frac{1}{2} \sum_{i,j=1}^N \sigma_i \sigma_j J_{i,j} = -\frac{1}{2N} \sum_{i,j=1}^N \sigma_i \sigma_j \sum_{\mu=1}^p \xi_i^\mu \xi_j^\mu$$

et la mesure de Gibbs associée est

$$\pi_{N,\beta}(\sigma) = \frac{\exp(-\beta H_N(\sigma))}{Z_{N,\beta}},$$

où

$$Z_N = \sum_{\sigma} e^{-\beta H_N(\sigma)}.$$

Nous supposons que les messages sont indépendants et tels que les v.a. $(\xi_i^\mu)_{i,\mu}$ sont i.i.d. de loi

$$\mathbb{P}(\xi_i^\mu = 1) = \mathbb{P}(\xi_i^\mu = -1) = \frac{1}{2}.$$

Remarquons que H_N est une fonction aléatoire dépendant de la réalisation de $\xi = (\xi_i^\mu)_{i,\mu}$ et que $\pi_{N,\beta}$ est donc une mesure aléatoire sur $\{-1, +1\}^N$. Il est connu que pour p ne dépendant pas N , le modèle de Hopfield admet une transition de phase en $\beta = 1$, tout comme le modèle de Curie Weiss. Lorsque $p = 1$, d'ailleurs, les deux modèles coïncident. Nous avons souhaité étendre le résultat de Madras et Zheng concernant l'algorithme du *swapping* pour le modèle de Curie Weiss ([**MaZ**]) au modèle de Hopfield et avons démontré le résultat suivant :

THÉORÈME 38 ([11]). *Considérons la chaîne de Markov \tilde{P} associée à l'algorithme du *swapping* avec M chaînes parallèles, pour simuler une réalisation de la mesure de Gibbs π_{N,β^*} associée au modèle de Hopfield avec $p = 2$ messages. Alors pour tout $\beta^* > 0$ et toutes réalisations de ξ^1 and ξ^2 telles que $\xi^1 \neq \pm \xi^2$, nous avons*

$$\text{Gap}(\tilde{P}) \geq \frac{1}{12} \frac{\vartheta^{4\frac{N}{M}+24}}{M^7 N^8}. \quad (8)$$

Ici $\vartheta = e^{-\beta^*/2}$. En particulier, si nous choisissons $M = cN$ avec une constante c positive, cela implique que l'algorithme du *swapping* est rapidement mélangeant.

REMARQUE 39. - *Minorer $\text{Gap}(\tilde{P})$ par l'inverse d'un polynôme en N assure une vitesse de convergence polynomiale. En effet, le lien entre $\text{Gap}(\tilde{P})$ et la vitesse de convergence de la chaîne vers la mesure d'équilibre est un résultat général (voir par exemple [**Si**] ou [11], Appendice A).*

- *Nous aurions aimé pouvoir considérer le cas p fini quelconque, mais les techniques utilisées ne sont pas facilement généralisables. Le cas où p dépend de N est encore plus complexe, les calculs dans le cas $p = 2$ montrant que la puissance de N et M dans la borne obtenue (8) dépend de p . A ma connaissance, il n'y a pas de résultat connu dans ce cas pour l'algorithme du *swapping*.*

8. Application au modèle de Blume-Emery-Griffiths

Introduisons le modèle de Blume-Emery-Griffiths. Pour un paramètre $K > 0$ fixé, la fonction énergie définie sur $\Omega = \{-1, 0, 1\}^N$ est donnée par

$$H(\sigma) = \sum_{j=1}^N \sigma_j^2 - \frac{K}{N} \left(\sum_{j=1}^N \sigma_j \right)^2,$$

pour $\sigma \in \Omega$. La mesure de Gibbs associée est alors

$$\pi_\beta(\sigma) = \frac{e^{-\beta H(\sigma)}}{Z(\beta)} = \frac{e^{-\beta H(\sigma)}}{\sum_{\sigma'} e^{-\beta H(\sigma')}} = \frac{e^{\beta(-\sum_{j=1}^N \sigma_j^2 + \frac{K}{N}(\sum_{j=1}^N \sigma_j)^2)}}{\sum_{\sigma'} e^{-\beta H(\sigma')}},$$

où $Z(\beta)$ est la constante de normalisation.

L'intérêt de ce modèle est qu'il présente une structure de transitions de phase très riche (voir [**EOT**] et [14]). La première région qui nous intéresse est la suivante : il

existe $K_c > K_{\text{low}} > 0$ tels que pour $K \in]K_{\text{low}}, K_c[$, le modèle admet une transition de phase discontinue en un $\beta_c^{(1)}(K)$ dépendant de K . Nous utilisons cette discontinuité pour montrer le résultat suivant :

THÉORÈME 40 ([14]). *Considérons le modèle de Blume-Emery-Griffiths avec $K \in]K_{\text{low}}, K_c[$. Alors pour $\beta > \beta_c^{(1)}(K)$, l'algorithme de simulated tempering converge en un temps exponentiel, puisqu'il existe un $c > 0$ tel que*

$$\text{Gap}(QP_{st}Q) \leq e^{-cN}.$$

REMARQUE 41. *Le théorème 37 implique un temps de convergence exponentiel pour l'algorithme de swapping dans cette région.*

Pour $K > K_c$, le modèle présente une transition de phase continue en un $\beta_c^{(2)}(K)$. Cela nous permet de montrer le résultat suivant :

THÉORÈME 42 ([14]). *Pour $K > K_c$ et $\beta > \beta_c^{(2)}(K)$, la chaîne de Markov QPQ associée à la méthode du swapping converge à une vitesse polynomiale pour le modèle de Blume-Emery-Griffiths, puisque*

$$\text{Gap}(QPQ) \geq \frac{1}{q(N)},$$

où q est un polynôme en N .

REMARQUE 43. *Le théorème 37 implique un temps de convergence polynomial pour l'algorithme de simulated tempering dans cette région.*

Quatrième partie

Etude probabiliste d'un algorithme de
classification lié à l'apprentissage profond

Cette partie correspond à l'article [21], réalisé avec Ghouti Hacène, Vincent Gripon et Matthias Löwe, et accessoirement aux chapitres 3 des ouvrages [19] et [20].

Les méthodes d'apprentissage statistique jouent un rôle croissant depuis quelques années dans de nombreux domaines. Il suffit de lire la presse qui se fait l'écho d'applications toujours plus nombreuses. Les applications des méthodes neuronales d'apprentissage statistique existent maintenant depuis plus de 30 ans, citons par exemple l'algorithme d'apprentissage d'un réseau de neurones par rétropropagation du gradient, qui date de 1985. Pour une introduction à l'apprentissage statistique, en particulier aux réseaux de neurones, voir par exemple le chapitre que j'ai écrit pour un livre sur le *Big Data* en assurance ([19] en français et [20] en anglais).

Cependant, ce n'est qu'assez récemment que l'Intelligence Artificielle basée sur des réseaux de neurones a été mentionnée de façon aussi importante dans les médias. Cela est dû au fait que l'augmentation très importante des moyens informatiques (puissance de calcul, capacité de stockage, mise à disposition de bases de données gigantesques) a permis la mise en oeuvre de modèles très performants. Un des progrès les plus importants dans ce domaine est probablement le concept d'apprentissage profond (*Deep Learning*), qui est une extension des réseaux de neurones classiques de type « Perceptron à une couche cachée » (voir [19] et [GoBC]). Les réseaux de neurones à apprentissage profond sont constitués de nombreuses couches successives et contiennent un grand nombre de paramètres à estimer. Leur calibrage nécessite de très grandes bases d'apprentissage et les algorithmes pour le faire deviennent très coûteux en temps de calcul. Il est aussi souvent nécessaire de procéder à des étapes d'hyperparamétrisation pour déterminer la structure du réseau la mieux adaptée aux données traitées.

C'est pour cette raison notamment que la technique de transfert d'apprentissage (*transfer learning*) a été développée : elle consiste à réaliser l'apprentissage d'un réseau de neurones sur une base de données conséquente, sur un ordinateur puissant et à utiliser le modèle obtenu pour traiter d'autres jeux de données, éventuellement sur des ordinateurs moins puissants. Les premières couches des réseaux de neurones de type *Deep Learning*, utilisés en traitement d'images notamment, sont le plus souvent des couches convolutives (voir [GoBC]), qui permettent d'extraire des caractéristiques des vecteurs en entrée. Ces informations sont recodées en un vecteur v de grande dimension (voir par exemple la partie expérimentation de notre article [21]), qui sera injecté en entrée des dernières couches du réseau de neurones. S'il s'agit d'une tâche de classification, les dernières couches du modèle ont pour rôle de prédire la classe du vecteur d'entrée à partir du vecteur v . Nous pouvons également utiliser des méthodes classiques de classification de type *k plus proches voisins* pour réaliser cette dernière tâche de classification. En analysant la distribution empirique de ces vecteurs v sur des cas réels, nous nous sommes rendus compte qu'ils ont une structure particulière et qu'une méthode de classification segmentée peut améliorer légèrement les performances en classification. Nous avons alors étudié mathématiquement la méthode proposée pour comprendre dans quel cas la segmentation peut en effet avoir un effet positif.

9. Un algorithme de classification avec segmentation

Le cadre mathématique est le suivant : considérons K classes $\mathcal{C}_1, \dots, \mathcal{C}_K \subseteq \mathbb{R}^d$, où nous supposons que la dimension d est élevée. Pour simplifier, supposons chaque classe constituée de M éléments. Choisissons c diviseur de d et écrivons $\mathbb{R}^d = \bigotimes_{j=1}^c \mathbb{R}^{d/c}$. Le $j^{\text{ème}}$ sous-espace sera noté $(\mathbb{R}^{d/c})_j$. Pour chaque j , considérons un dictionnaire D_j de nK vecteurs de taille d/c , tels que D_j contienne n vecteurs de chaque classe \mathcal{C}_k choisis au hasard (selon la loi uniforme). Pour cela, nous écrivons chaque $w \in \mathcal{C}_k$, $k = 1, \dots, K$ comme $w = w_1 \circ \dots \circ w_c$, où $w_j \in (\mathbb{R}^{d/c})_j$ et \circ représente la concaténation. Pour chaque classe \mathcal{C}_k , $k = 1, \dots, K$, nous tirons au hasard sans remplacement n segments de mots w_j dans \mathcal{C}_k , qui sont placés dans le dictionnaire D_j . Etant donné un nouveau mot z , nous le partitionnons de la même façon $z = z_1 \circ \dots \circ z_c$. Alors pour chaque $j = 1, \dots, c$ nous déterminons

$$\tilde{w}_j = \operatorname{argmin}\{\|w_j - z_j\|_{d/c} : w_j \in D_j\}. \quad (9)$$

Ici $\|\cdot\|_{d/c}$ est la distance Euclidienne dans $\mathbb{R}^{d/c}$. Définissons la variable U_j qui prend la valeur k , si \tilde{w}_j est le $j^{\text{ème}}$ segment d'un mot $w \in \mathcal{C}_k$. Si plusieurs mots minimisent la distance dans (9), U_j prend avec équiprobabilité une des valeurs des classes correspondant à ces mots. Finalement, nous posons

$$\chi(z) = \operatorname{argmax}_k \sum_{j=1}^c \{\mathbb{I}_{U_j=k}, k = 1, \dots, K\},$$

i.e. $\chi(z)$ est la classe la plus souvent trouvée pour les c segments (en cas d'égalité, nous procédons de nouveau à un tirage au sort). La classification par c -segmentation assigne à z la classe $\chi(z)$.

10. Une situation où la segmentation n'est pas bénéfique

Nous commençons par un exemple qui montre que les données doivent avoir une structure particulière pour que la méthode de classification par segmentation soit intéressante. Supposons que nous avons seulement deux classes \mathcal{C}_1 et \mathcal{C}_2 , construites de la façon suivante : deux vecteurs $m_1, m_2 \in \{-1, +1\}^d$ sont choisis au hasard et serviront de « vecteurs de base » des deux classes. Soit $0 < \rho < \frac{1}{2}$. Alors pour $\mu = 1, \dots, M$, soient $Y^\mu, \tilde{Y}^\mu \in \{-1, +1\}^d$ des vecteurs aléatoires i.i.d. ayant des coordonnées i.i.d. telles que $\mathbb{P}(Y_1^1 = 1) = 1 - \rho$ et $\mathbb{P}(Y_1^1 = -1) = \rho$. Nous définissons $\mathcal{C}_1 = \{Y^\mu \times m_1, \mu = 1, \dots, M\}$ et $\mathcal{C}_2 = \{\tilde{Y}^\mu \times m_2, \mu = 1, \dots, M\}$. Ici la multiplication se fait coordonnée par coordonnée, i.e. $(Y^\mu \times m_1)_i = Y_i^\mu m_{1,i}$. Soit $c \in \mathbb{N}$ diviseur de d . Pour chaque $1 \leq j \leq c$, pour faciliter les calculs, nous prenons un dictionnaire D_j constitué seulement de deux segments, l'un, w_j^1 , appartenant à la classe \mathcal{C}_1 , et l'autre, w_j^2 , appartenant à la classe \mathcal{C}_2 . Dans ce cas, nous montrons que la méthode par c -segmentation ne permet pas d'améliorer la précision obtenue avec la méthode Euclidienne simple.

PROPOSITION 44. *Dans la situation décrite ci dessus, supposons que c et d/c sont impairs (pour éviter d'avoir recours au tirage au sort) et que w est distribué selon la même loi qu'un mot de la classe \mathcal{C}_1 (et indépendant de tous les mots de toutes les classes).*

Alors il existe un nombre $I > 0$ tel que

$$\lim_{d \rightarrow \infty} \frac{1}{d} \log \mathbb{P}(\chi(w) = 2 \text{ avec } c = 1) = -I,$$

tandis que

$$\lim_{d \rightarrow \infty} \frac{1}{d} \log \mathbb{P}(\chi(w) = 2 \text{ avec } 1 \ll c \ll d) = -I/2.$$

REMARQUE 45. - Cette proposition montre que la méthode avec c -segmentation n'améliore pas les performances obtenues avec la méthode sans segmentation ($c = 1$). Nous pouvons remarquer quand même que la méthode avec c -segmentation avec $1 \ll c \ll d$ fournit la bonne classe avec une probabilité proche de 1.

- Il est aussi possible de montrer que la méthode avec $c = 1$ est plus performante qu'une autre méthode « naturelle », la méthode de comparaison bit par bit ($c = d$).

11. Des situations où la segmentation est bénéfique

Nous allons présenter dans cette section des cas où la méthode de classification avec c -segmentation avec $1 \ll c \ll d$ a un intérêt. Commençons par décrire un cas simple qui pourra donner lieu ensuite à des extensions. En fait, cet exemple est inspiré par des observations sur des données réelles qui montrent que les vecteurs v obtenus en pré-sortie des réseaux de neurones utilisés ont quelques coordonnées beaucoup plus élevées que la majorité des valeurs observées.

Définissons $K = 2$ classes \mathcal{C}_1 et \mathcal{C}_2 par

$$\mathcal{C}_1 = \{Y^\mu + m_1, \mu = 1, \dots, M\} \quad \text{et} \quad \mathcal{C}_2 = \{\tilde{Y}^\mu + m_2, \mu = 1, \dots, M\}.$$

Ici, pour simplifier, nous choisissons $m_1 = (0, \dots, 0)$ et $m_2 = (1, \underbrace{0, \dots, 0}_{l-1 \text{ fois}}, 1, \underbrace{0, \dots, 0}_{l-1 \text{ fois}}, 1 \dots 0)$,

i.e. m_2 a des « 1 » à des positions régulières. De plus, nous considérons des vecteurs Y^μ et \tilde{Y}^μ i.i.d dans \mathbb{R}^d , de coordonnées indépendantes telles que $\mathbb{P}(Y_1^1 = N) = \mathbb{P}(\tilde{Y}_1^1 = N) = p = 1 - \mathbb{P}(Y_1^1 = 0) = 1 - \mathbb{P}(\tilde{Y}_1^1 = 0)$, où $N > 0$ et $p \in]0, 1[$ seront choisis ultérieurement.

Soit c fixé. Pour chaque segment $1 \leq j \leq c$, nous construisons de nouveau un dictionnaire D_j constitué d'un segment w_j^1 provenant d'un mot de \mathcal{C}_1 et d'un segment w_j^2 d'un mot de \mathcal{C}_2 . Supposons de nouveau que nous souhaitons classer un mot w de même loi de probabilité qu'un mot de \mathcal{C}_1 (mais indépendant de tous les mots de toutes les classes). La première observation est que pour l et p assez petits, la méthode de comparaison coordonnée par coordonnée échoue :

PROPOSITION 46. *Supposons $l \leq d^{1/4}$, $N > 0$ et $p \leq \frac{1}{l}$. Alors pour $c = d$,*

$$\mathbb{P}(\chi(w) = 1) \rightarrow \frac{1}{2} \quad \text{lorsque } d \rightarrow \infty. \quad (10)$$

REMARQUE 47. *La valeur obtenue dans l'équation (10) correspond bien entendu au pire des cas, dans le cas de deux classes.*

La méthode simple basée sur la distance Euclidienne entre les vecteurs ($c = 1$) échoue également pour de larges plages de valeurs des paramètres :

PROPOSITION 48. *Si $l \rightarrow \infty$ et $p \gg \max(\frac{1}{d}, \frac{1}{N^2l})$, alors pour $c = 1$*

$$\mathbb{P}(\chi(w) = 1) \rightarrow \frac{1}{2} \quad \text{lorsque } d \rightarrow \infty. \quad (11)$$

Existe-t-il une valeur de c telle que la méthode par c -segmentation ait de bonnes performances ? Nous pouvons répondre par l'affirmative :

PROPOSITION 49. *Si $l \ll \frac{1}{p}$ et $c = d/l$, alors*

$$\mathbb{P}(\chi(w) = 1) \rightarrow 1, \quad \text{lorsque } l \rightarrow \infty \text{ et } d \rightarrow \infty.$$

Nous pouvons regrouper ces résultats sous la forme suivante :

THÉORÈME 50. *Supposons que $d, l \rightarrow \infty$, $l \leq d^{\frac{1}{4}}$, $p \ll \frac{1}{l}$, mais $p \gg \max(\frac{1}{d}, \frac{1}{N^2l})$, alors pour $c = d$ et $c = 1$ nous avons*

$$\mathbb{P}(w \text{ est mal classé}) \rightarrow 1/2,$$

tandis que pour $c = d/l$, nous avons

$$\mathbb{P}(w \text{ est mal classé}) \rightarrow 0.$$

Après avoir étudié ce premier modèle élémentaire, qui met en valeur l'intérêt de la méthode avec segmentation, nous allons présenter des variantes un plus riches de notre modèle. Une première extension possible est le cas à plus de 2 classes. De manière similaire au cas à 2 classes, nous pouvons considérer jusqu'à $l + 1$ classes, où $\mathcal{C}_1 = \{Y^{1,\mu} + m_1, \mu = 1, \dots, M\}$ avec $m_1 = 0$ et pour $k = 2, \dots, K$, $\mathcal{C}_k = \{Y^{k,\mu} + m_k, \mu = 1, \dots, M\}$. Les vecteurs $(Y^{k,\mu})$ sont des vecteurs aléatoires i.i.d. dans \mathbb{R}^d , de coordonnées indépendantes telles que $\mathbb{P}(Y_i^{k,\mu} = N) = p = 1 - \mathbb{P}(Y_i^{k,\mu} = 0)$ et les vecteurs $(m_k)_{k=2,\dots,K}$ sont des concaténations de chaînes de longueur l , telles que chacune de ces chaînes ait un seul « 1 » et les autres coordonnées nulles, et les « 1 » sont placés à différentes positions pour les différents m_k . Dans ce cas plus général, le Théorème 50 devient :

THÉORÈME 51. *Dans la situation à K classes décrite ci-dessus, supposons que $d, l \rightarrow \infty$, $l \leq d^{\frac{1}{4}}$, $p \ll \frac{1}{l}$, mais $p \gg \max(\frac{1}{d}, \frac{1}{N^2l})$. Alors pour $c = d$ et $c = 1$ nous avons*

$$\mathbb{P}(w \text{ est mal classé}) \rightarrow 1/K$$

tandis que pour $c = d/l$, nous avons

$$\mathbb{P}(w \text{ est mal classé}) \rightarrow 0.$$

Ceci montre aussi que le cas à deux classes est générique. Nous allons donc nous limiter à ce cas pour les variantes qui suivent.

Jusqu'à présent, nous avons considéré que les dictionnaires sont constitués d'un seul segment de chaque classe. Nous allons maintenant étudier l'influence d'un dictionnaire plus grand. Supposons maintenant que les dictionnaires D_j contiennent ν segments de chaque classe, i.e. $D_j = \{w_j^{1,1}, \dots, w_j^{1,\nu}, w_j^{2,1}, \dots, w_j^{2,\nu}\}$ et que les mots sont toujours de la forme $w^{1,\mu} = m_1 + Y^{1,\mu}$, $w^{2,\mu} = m_2 + Y^{2,\mu}$. Nous pouvons montrer dans ce cas que les Propositions 46 à 49 restent vraies.

Dans l'exemple considéré, les propriétés des données qui font que la méthode par c -segmentation fonctionne mieux que les méthodes standard sont les suivantes :

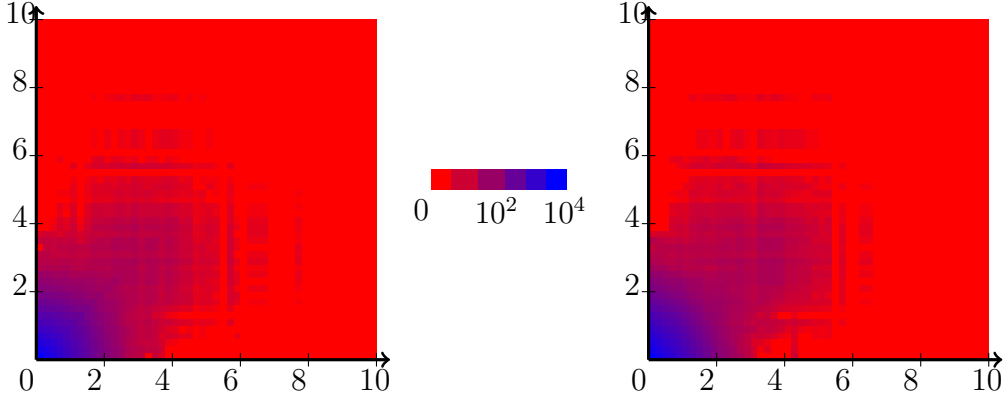


FIGURE 1. Histogramme des valeurs des vecteurs de pré-sortie pour les données CIFAR10 et le réseau InceptionV3. Même classe (à gauche), différentes classes (à droite).

- Les vecteurs de chaque classe sont obtenus par des perturbations *rare*s mais *élevées* d'un vecteur de base. La majorité des coordonnées des vecteurs de base de deux classes distinctes coïncident.
- La fréquence des perturbations est beaucoup plus faible que la fréquences des coordonnées où les vecteurs de base diffèrent.

Cependant, en analysant les données utilisées pour nos applications numériques, nous avons constaté que notre modèle décrit bien le comportement d'une classe, mais pas de deux classes simultanément. En effet, les pixels qui ont une valeur élevée pour une classe ont aussi très probablement une valeur élevée pour une autre classe (avec une variance plus élevée pour les pixels qui prennent des valeurs élevées). Ceci est visible sur la Figure 1 : pour illustration, nous considérons les vecteurs v obtenus pour les images de la base CIFAR10 avec le réseau de neurones « Inception V3 » et nous traçons l'historgramme des valeurs prises par couples quand nous prenons les mêmes coordonnées de deux vecteurs d'une même classe ou de deux classes différentes.

Pour prendre ceci en compte, nous modifions le modèle de la façon suivante :

$$\mathcal{C}_1 = \{N^\mu Y + m_1, \mu = 1, \dots, M\} \quad \text{et} \quad \mathcal{C}_2 = \{\tilde{N}^\mu Y + m_2, \mu = 1, \dots, M\}.$$

De nouveau $m_1 = (0, \dots, 0)$ et $m_2 = (1, \underbrace{0, \dots, 0}_{l-1 \text{ fois}}, 1, \underbrace{0, \dots, 0}_{l-1 \text{ fois}}, 1 \dots 0)$. De plus, le vecteur Y , qui est le même pour les deux classes, est un vecteur de variables aléatoires i.i.d. de loi de Bernoulli de paramètre p , i.e. $\mathbb{P}(Y_1 = 1) = p = 1 - \mathbb{P}(Y_1 = 0)$. Finalement, les N^μ et les \tilde{N}^μ sont des vecteurs aléatoires i.i.d. à valeurs dans \mathbb{R}^d avec des coordonnées i.i.d., telles que $\mathbb{P}[N_1^\mu \geq a] = 1$, pour un certain $a > 0$. Nous considérons des dictionnaires D_j constitués d'un seul segment w_j^1 et w_j^2 respectivement de chacune des classes et nous souhaitons estimer la classe d'un mot w de même loi qu'un mot de \mathcal{C}_1 (mais indépendant de tous les mots de toutes les classes). Supposons que $w^1 = YN$, $w_2 = m_2 + Y\tilde{N}$ et $w = Y\bar{N}$. Nous avons alors le résultat suivant :

THÉORÈME 52. *Dans le modèle décrit ci-dessus, supposons que $d, l \rightarrow \infty$, $l \leq d^{\frac{1}{4}}$, $p \ll \frac{1}{l}$, mais $p \gg \max(\frac{1}{d}, \frac{1}{d^2 l})$. Alors pour $c = d$ et $c = 1$*

$$\mathbb{P}(w \text{ est mal classé}) \rightarrow 1/2$$

tandis que pour $c = d/l$,

$$\mathbb{P}(w \text{ est mal classé}) \rightarrow 0.$$

Travaux de l'auteur

- [1] Franck Vermet. Convergence de la variance de l'énergie libre pour le modèle de Hopfield. *C. R. Acad. Sci. Paris Sér. I Math.*, 315(9) :1001–1004, 1992.
- [2] Franck Vermet. Discrétisation d'une équation différentielle stochastique dont les coefficients ne dépendent pas du temps et calcul approché d'espérances de fonctionnelles de la solution. In *Fascicule de probabilités*, volume 1992 of *Publ. Inst. Rech. Math. Rennes*, page 65. Univ. Rennes I, Rennes, 1992.
- [3] Franck Vermet. *Etude asymptotique d'un réseau neuronal : le modèle de mémoire associative de Hopfield*. PhD thesis, Université de Rennes 1, 1994.
- [4] Franck Vermet. Transition de phase et vitesse de fuite pour une mesure discrète de Edwards non symétrique sur \mathbf{Z} . *C. R. Acad. Sci. Paris Sér. I Math.*, 322(6) :567–570, 1996.
- [5] Franck Vermet. Phase transition and law of large numbers for a non-symmetric one-dimensional random walk with self-interactions. *J. Appl. Probab.*, 35(1) :55–63, 1998.
- [6] Matthias Löwe and Franck Vermet. The storage capacity of the Blume-Emery-Griffiths neural network. *J. Phys. A*, 38(16) :3483–3503, 2005.
- [7] Matthias Löwe and Franck Vermet. The storage capacity of the Hopfield model and moderate deviations. *Statist. Probab. Lett.*, 75(4) :237–248, 2005.
- [8] Remco van der Hofstad, Matthias Löwe, and Franck Vermet. The effect of system load on the existence of bit errors in CDMA with and without parallel interference cancelation. *IEEE Trans. Inform. Theory*, 52(10) :4733–4741, 2006.
- [9] Matthias Löwe and Franck Vermet. The capacity of q -state Potts neural networks with parallel retrieval dynamics. *Statist. Probab. Lett.*, 77(14) :1505–1514, 2007.
- [10] Matthias Löwe and Franck Vermet. Capacity bounds for the CDMA system and a neural network : a moderate deviations approach. *ESAIM Probab. Stat.*, 13 :343–362, 2009.
- [11] Matthias Löwe and Franck Vermet. The swapping algorithm for the Hopfield model with two patterns. *Stochastic Process. Appl.*, 119(10) :3471–3493, 2009.
- [12] Matthias Löwe and Franck Vermet. The Hopfield model on a sparse Erdős-Renyi graph. *J. Stat. Phys.*, 143(1) :205–214, 2011.
- [13] Corwin Wright, Robert Scott, Darran Furnival, Pierre Ailliot, and Franck Vermet. Global Observations of Ocean-bottom Subinertial Current Dissipation. *J. Phys. Oceanography*, 43(2) :402–413, 2013.
- [14] Mirko Ebbers, Holger Knöpfel, Matthias Löwe, and Franck Vermet. Mixing times for the swapping algorithm on the Blume-Emery-Griffiths model. *Random Structures Algorithms*, 45(1) :38–77, 2014.
- [15] Matthias Löwe and Franck Vermet. Capacity of an associative memory model on random graph architectures. *Bernoulli*, 21(3) :1884–1910, 2015.
- [16] Judith Heusel, Matthias Löwe, and Franck Vermet. On the capacity of an associative memory model based on neural cliques. *Statist. Probab. Lett.*, 106 :256–261, 2015.
- [17] Vincent Gripon, Judith Heusel, Matthias Löwe, and Franck Vermet. A comparative study of sparse associative memories. *J. Stat. Phys.*, 164(1) :105–129, 2016.
- [18] Mete Demircigil, Judith Heusel, Matthias Löwe, Sven Upgang, and Franck Vermet. On a model of associative memory with huge storage capacity. *J. Stat. Phys.*, 168(2) :288–299, 2017.

- [19] Emmanuel Berthel , Romain Billot, C cile Bothorel, Marine Habart, Jacques Janssen, Philippe Lenca, Florence Picard, Gilbert Saporta, and Franck Vermet. *Le big data pour les compagnies d'assurance*. ISTE  ditions, 2017.
- [20] Emmanuel Berthel , Romain Billot, C cile Bothorel, Marine Habart, Jacques Janssen, Philippe Lenca, Florence Picard, Gilbert Saporta, and Franck Vermet. *Big Data for Insurance companies*. Wiley-ISTE, 2018.
- [21] Vincent Gripon, Ghouthi Hacene, Matthias L we, and Franck Vermet. Improving Accuracy of Non-parametric Transfer Learning via Vector Segmentation. In *Conference IEEE ICASSP 2018*, pages 2966–2970, 2018.
- [22] Vincent Gripon, Matthias L we, and Franck Vermet. Associative memories to accelerate approximate nearest neighbor search. *Appl. Sci.*, 8(9) :1676, 2018.

Bibliographie

- [AGS] D. Amit, H. Gutfreund, H. Sompolinsky, Spin-glass models of neural networks, *Phys. Rev. A (3)* **32** No 2 : 1007–1018 (1985).
- [Am] S. Amari, Characteristics of sparsely encoded associative memory, *Neural Networks* **2** No 6 : 451–457 (1989).
- [AY] S. Amari, H.-F. Yanai, Auto-associative memory with two-stage dynamics of nonmonotonic neurons, *IEEE Trans. Neural Networks* **7** : 803–815 (1996).
- [BhR] N. Bhatnagar, D. Randall, Torpid mixing of simulated tempering on the Potts model, in : Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms, ACM, New York, 478–487 (2004),
- [Bov] A. Bovier, Sharp upper bounds on perfect retrieval in the Hopfield model, *J. Appl. Probab.* **36** No 3 : 941–950 (1999).
- [Bur] D. Burshtein, Nondirect convergence radius and number of iterations of the Hopfield associative memory, *IEEE Trans. Inform. Theory* **40** No 3 : 838–847 (1994).
- [CL] F. Chung, L. Lu, Connected components in random graphs with given expected degree sequences, *Ann. Comb.* **6** No 2 : 125–145 (2002).
- [CR] F. Chung, M. Radcliffe, On the spectra of general random graphs, *Electron. J. Combin.* **18** No 1 : Paper 215, 14 (2011).
- [De] P. de Jong, A central limit theorem for generalized multilinear forms, *J. Multiv. Anal.* **34** : 275–289 (1990).
- [EL] M. Ebbers, M. Löwe, Torpid mixing of the swapping chain on some simple spin glass models, *Markov Process. Relat. Fields* **15** No 1 : 59–80 (2009).
- [EOT] R. Ellis, P. Otto, H. Touchette, Analysis of phase transitions in the mean-field Blume-Emery-Griffiths model, *Ann. Appl. Probab.* **15** No 3 : 2203–2254 (2005).
- [FHK] A. Fey, R. Van der Hofstad, M. Klok, Large deviations for eigenvalues of sample covariance matrices, with applications to mobile communication systems, *Adv. in Appl. Probab.* **40** No 4 : 1048–1071 (2008).
- [Ge] C. Geyer, Markov chain Monte Carlo maximum likelihood, in : Computing Science and Statistics : Proceedings of 23rd Symposium on the Interface Interface Foundation, Fairfax Station, Virginia, 156–163 (1991),
- [GoBC] I. Goodfellow, Y. Bengio and A. Courville, Deep Learning, MIT Press, (2016) <http://www.deeplearningbook.org> ,
- [GrB] V. Gripon, C. Berrou, Sparse neural networks with large learning diversity, *IEEE Transactions on Neural Networks* **22** No 7 : 1087–1096 (2011).
- [HK1] R. Van der Hofstad, M. Klok, Performance of DS-CDMA systems with optimal hard-decision parallel interference cancellation, *IEEE Trans. Inform. Theory* **49** No 11 : 2918–2940 (2003).
- [HK2] R. Van der Hofstad, M. Klok, Improving the performance of third-generation wireless communication systems, *Adv. in Appl. Probab.* **36** No 4 : 1046–1084 (2004).
- [Hopf] J.J., Hopfield, Neural networks and physical systems with emergent collective computational abilities, *Proc. Nat. Acad. Sci. U.S.A.* **79** No 8 : 2554–2558 (1982).

- [KH] D. Krotov and J. Hopfield, Dense Associative Memory for Pattern Recognition, *Advances in Neural Information Processing Systems* **29**, D. D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, R. Garnett, 1172–1180 (2016).
- [KP1] J. Komlós, R. Paturi, Convergence results in an associative memory model, *Neural Networks* **1** No 3 : 239–250 (1988).
- [KP2] J. Komlós, R. Paturi, Effect of connectivity in an associative memory model, *J. Comput. System Sci.* **47** No 2 : 350–373 (1993).
- [MacE] R. McEliece, E. Posner, E. Rodemich, S. Venkatesh, The capacity of the Hopfield associative memory, *IEEE Trans. Inform. Theory* **33** No 4 : 461–482 (1987).
- [MaP] N. Madras, Neil, M. Piccioni Importance sampling for families of distributions, *Ann. Appl. Prob.* **9** No 4 : 1202–1225 (1999).
- [MaZ] N. Madras, Z. Zheng On the swapping algorithm, *Random Struct. Algorithms* **22** No 1 : 66–97 (2003).
- [MP] E. Marinari, G. Parisi, Simulated tempering : A new Monte Carlo scheme, *Europhys Lett.* **19** No 6 : 451–458 (1992).
- [New] C. Newman, Memory capacity in neural network models : Rigorous lower bounds, *Neural Networks* **1** No 3 : 223–238 (1988).
- [Si] A. Sinclair, Algorithms for random generation and counting : a Markov chain approach, Birkhauser Verlag, Basel (1993).
- [WBL-H] D. Willshaw, O. Buneman, H. Longuet-Higgins, Non-Holographic Associative Memory, *Nature* **222** : 960–962 (1969).
- [Zh] Z. Zheng, On swapping and simulated tempering algorithms, *Stochastic Process. Appl.* **104** No 1 : 104–154 (2003).