

Redire que l'étude théorique a été faite "à prof-deur"  
sur 1D pour des raisons de lecture facilitée, et qu'il s'agit  
de jeter un œil sur la 2D?

À recommander?

## Chapitre 7

# Extension aux cartes en deux dimensions

### Sommaire

<b>7.1 Présentation de l'expérience</b>	1
7.1.1 Modèle d'entrées	1
7.1.2 Architecture	2
<b>7.2 Résultats</b>	4
7.2.1 Organisation des cartes sur des entrées dépendantes	4
7.2.2 Convergence des poids et motifs de poids contextuels	5
7.2.3 Convergence de la relaxation	7
7.2.4 Prédiction d'entrée	10
7.2.5 Organisation de cartes sur des entrées indépendantes	11
<b>7.3 Conclusion</b>	14

Nous avons étudié des comportements de base de cartes 1D sur des données en une dimension. Cependant, les cartes auto-organisatrices utilisées en pratique sont des cartes en deux dimensions. Elles apportent en effet une meilleure qualité de quantification vectorielle sur des données de plus grande dimension, tout en restant peu coûteuses en calcul. Ce chapitre présente donc quelques résultats obtenus sur des architectures CxSOM construites sur des cartes en deux dimensions. Nous cherchons à étudier ici si les mécanismes d'organisation observés sur des cartes 1D sont transposables à des cartes en deux dimensions, à savoir :

- La qualité de la quantification vectorielle des entrées externes,
- L'apprentissage du modèle par chacune des cartes,
- L'organisation des poids contextuels,
- La convergence de la relaxation.

Ceci nous permettra d'identifier les limites et perspectives du passage des cartes 1D à 2D dans des architectures.

## 7.1 Présentation de l'expérience

### 7.1.1 Modèle d'entrées

Nous évaluerons dans ce chapitre le comportement de cartes prenant des entrées 2D dans un cas géométrique similaire aux cartes 1D du chapitre précédent prenant des entrées unidimensionnelles.

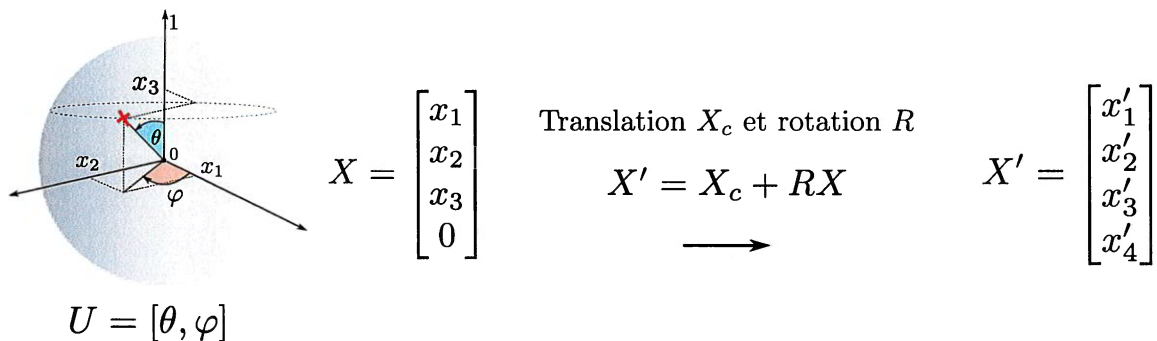


FIGURE 7.1 – Transformation d'une surface d'un espace 3D vers un espace 4D (ou 6D dans la section suivante). Les points restent positionnés sur une surface, mais sont plongés dans un espace de plus grande dimension. La rotation permet de répartir les coordonnées des points sur les quatre dimensions sans modifier la structure des entrées.

Il est inutile d'étudier des cartes deux dimensions prenant des entrées en une dimension ; nous prendrons donc comme entrées multimodales des points en 4 dimensions pour une structure de 2 cartes. Chacune des cartes prend donc une entrée 2D. Nous choisissons ces points sur une sphère en 3D dont on a effectuée une rotation en dimension 4 dans l'espace 4D, de la même façon que les entrées en 3D se trouvaient sur un cercle 2D dont on a effectué une rotation 3D dans l'espace, voir figure 7.1. De cette manière, les points sont situés sur une surface dans l'espace en 4D. La rotation permet de changer le repère et de répartir les valeurs sur quatre coordonnées au lieu de trois. La translation place le centre en  $[0.5, 0.5, 0.5, 0.5]$  afin que les coordonnées soient comprises entre 0 et 1. La rotation en quatre dimensions est générée en multipliant 6 matrices de rotations élémentaires laissant chacune un des plans de la base 4D fixe, en choisissant des angles répartissant bien les valeurs sur les quatre axes. La rotation est ici un moyen de représenter la surface étudiée sur les quatre dimensions au lieu de trois, sans modifier  $U$ . La variable  $U$  paramétrant cette surface est en deux dimensions, il s'agit des angles  $\theta$  et  $\varphi$  de représentation polaire de la sphère et est une représentation du modèle d'entrées. Nous normalisons les entrées entre 0 et 1 indépendamment sur chaque dimension, entraînant une légère déformation de la sphère.

Les entrées fournies à chaque carte sont ensuite  $X^{(1)} = [x_0, x_1]$  et  $X^{(2)} = [x_2, x_3]$ . Par symétrie, l'agencement des coordonnées importe peu. Les paires  $X^{(1)}, X^{(2)}$  correspondant à un même point sont présentées au même moment à l'architecture.

### 7.1.2 Architecture

L'architecture de cartes en deux dimensions est construite sur le même principe qu'une architecture de cartes 1D, le modèle présenté au chapitre ?? étant valable pour n'importe quelle dimension de carte. Nous nous intéressons d'abord à une architecture de deux cartes. Les nœuds sont positionnées sur une grille carrée et indexés entre 0 et 1 sur chaque dimension, que nous noterons  $\mathbf{p}^{(i)} = (p_0^{(i)}, p_1^{(i)})$  sur les figures. Chaque carte possède donc une couche de poids externes  $\omega_e \in [0, 1]^2$  et une couche de poids contextuels  $\omega_c \in [0, 1]^2$ . Les poids contextuels représentent les positions des BMUs de l'autre cartes qui sont les positions  $\mathbf{p}^{(2)}$  pour la carte  $M^{(1)}$  et  $\mathbf{p}^{(1)}$  pour la carte  $M^{(2)}$ . Nous prenons des cartes de taille  $100 \times 100$ . Les activités  $a_e$  et  $a_c$  sont calculées par une activation

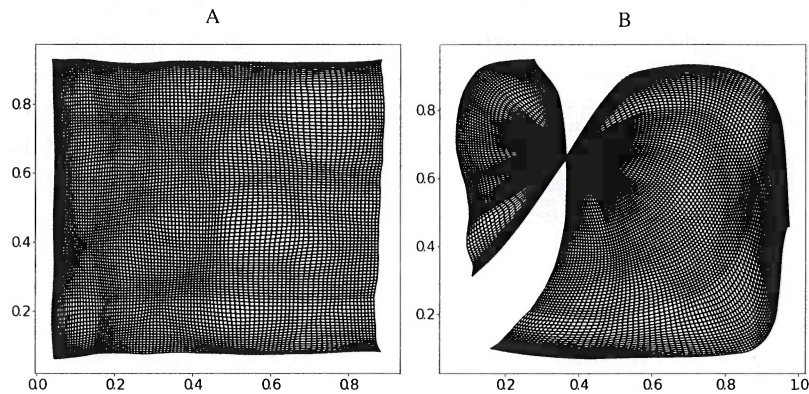



FIGURE 7.2 – Exemple de point de torsion dans une carte 2D dépliée sur des entrées dans  $[0, 1]^2$  (B). La carte A au contraire est bien dépliée : deux entrées proches sont représentées par des BMUs proches. Cette disposition présentant un point de torsion peut évoluer vers une carte bien dépliée ou vers un état stable présentant un point de torsion, en fonction des paramètres d'apprentissage.

gaussienne, qui est la même qu'en dimension 1.

$$a_e(X, \mathbf{p}) = \exp\left(\frac{\|X - \omega_e(\mathbf{p})\|^2}{2\sigma^2}\right)$$

de même pour  $a_c$ . La norme utilisée est ici la norme euclidienne, que ce soit dans l'espace d'entrée et dans l'espace des positions de la carte. Les fonctions de voisinage sont également définies à partir de la norme euclidienne dans l'espace des positions d'une carte.

La notion d'organisation dans une carte 2D est plus difficile à formuler que dans une carte en une dimension. Par exemple, la topologie intuitive d'une carte "organisée" sur le carré  $[0, 1]^2$  est que les quatre coins de la carte viennent se placer dans les coins du carré. Cependant, les mécanismes d'organisation peuvent conduire à une carte tordue en son milieu, par exemple en  Cette organisation respecte la continuité des poids au sens des cartes auto-organisatrices : deux poids proches dans la carte correspondent à des entrées proches, peut-être une configuration stable des poids, mais n'est pas forcément la topographie représentant le mieux l'espace d'entrée. De plus, l'organisation et la convergence des poids externes n'est pas assurée pour un rayon de voisinage constant, même dans une carte de Kohonen classique. Des travaux anciens ont prouvé la convergence des SOM en une dimension (Cottrell et al. 1998) mais seulement partiellement pour les cartes en deux dimensions, même avec des paramètres d'apprentissage décroissant (Flanagan 1996). En pratique, la convergence des poids des SOMs en deux dimensions est observée avec un rayon de voisinage décroissant, mais pas de travaux à notre connaissance n'ont étudié le comportement des SOMs 2D lorsque les rayons restent constants. La convergence des poids des cartes est un enjeu crucial en deux dimensions.

Dans une architecture CxSOM, cela pourrait poser un problème pour l'utilisation de la position du BMU en tant que représentation de l'entrée : deux entrées proches peuvent avoir leur BMU de chaque côté de la carte. Dans un premier temps, nous avons choisi de favoriser le bon dépliement des poids externes en modifiant l'initialisation des poids, auparavant aléatoire. Pour cela, nous laissons les poids externes s'organiser sur 500 itérations à partir des activités externes et sans prendre en compte les poids contextuels, en prenant un grand rayon de voisinage  $r_e = 0.5$ . Ce grand rayon

en gros la CxSOM 2D est une SOM 2D avec des limites en terme d'organisation? → Mais ça peut poser des pb spécifiques aux CxSOM?



permet d'éviter les zones de torsion dans la carte et pré-répartit les poids externes. Après cette étape préalable, nous réduisons le rayon externe à  $r_e = 0.2$  et effectuons l'apprentissage des poids externes et contextuels comme décrit dans le modèle CxSOM. Les poids externes affinent alors leur apprentissage et les poids contextuels s'organisent sur une carte "bien dépliée". La généralisation de l'organisation de CxSOM sur des carte "mal dépliées" sera à explorer pour assurer une robustesse de l'algorithme sur des données quelconques : il est en effet compliqué de s'assurer du bon dépliement d'une carte en pratique sur des données de grande dimension. Nous laissons cette question à des travaux futurs.

## 7.2 Résultats

### 7.2.1 Organisation des cartes sur des entrées dépendantes

Nous étudions d'abord l'organisation des cartes sur les données prises sur la sphère 3D pivotée dans un espace en 4D décrite en figure 7.1. Les données ont ici deux degrés de liberté qui sont les angles paramétrant la représentation polaire de la sphère,  $U$  est ainsi en deux dimensions. Nous prendrons dans une première expérience les paramètres  $r_c = 0.02$  et  $r_e = 0.2$ . Nous nous intéressons en premier lieu à l'organisation des poids externes et contextuels des cartes. Sur des cartes 1D, nous avons observé que les poids externes s'organisent comme une carte classique, tandis que les poids contextuels forment des zones.

La figure 7.3, en haut, présente la disposition des poids externes et contextuels de l'architecture de deux cartes après 300 000 itérations d'apprentissage. Nous représentons les poids externes dans l'espace des entrées : un nœud de la carte est positionné en  $\omega_e(\mathbf{p})$  qui est une position 2D. La figure montre que chaque carte représente ses entrées externes de la même façon qu'une carte classique, ce qui est bien attendu du modèle CxSOM. Chacune des cartes s'étend sur le disque dans lequel sont tirées leurs entrées externes. En bas, nous représentons les poids contextuels de la carte sous forme de carte de coloration. Un pixel positionné en position  $\mathbf{p}$  est coloré selon la valeur de son poids  $\omega_c(\mathbf{p})$ , une valeur en deux dimensions. Ces tracés font apparaître des motifs alternés dans chaque carte qui rappellent les motifs présents en une dimension. Le comportement observé en 1D se retrouve donc sur cet exemple de carte 2D. Une première différence avec les cartes en une dimension est la forme des zones, moins régulières. Comme en une dimension, ces zones dépendent des rayons de voisinage  $r_c$  et  $r_e$ . Nous verrons leur influence plus bas.

Dans un deuxième temps, nous nous intéressons à l'apprentissage du modèle  $U$  dans chacune des cartes. Comme en 1D, nous pouvons tracer la valeur de  $U$  selon la position du BMU dans chaque carte sur une étape de test lancée sur la configuration des poids après apprentissage. Cette étape de test comporte 10000 points. Ces tracés permettent de faire figurer les zones mortes de la carte. En une dimension, nous avons noté que les zones de BMU correspondent aux zones des poids contextuels ; deux zones proches codent pour des valeurs de  $X$  proches mais pour des  $U$  différents. Dans chaque carte,  $U$  est alors une fonction du BMU : cette propriété marque l'apprentissage du modèle par chacune des cartes.

La figure 7.4 présente les valeurs de la variable 2D  $U$ , représentée par la couleur des points, en fonction de la position du BMU  $\Pi^{(1)}$  et  $\Pi^{(2)}$  de chaque carte. Nous y faisons figurer deux points en rouge et bleu : ces deux entrées ont la même valeur de  $X^{(1)}$  mais une valeur différente de  $X^{(2)}$  et donc  $U$ . Nous voyons sur la figure que les comportements observés en 1D se reproduisent dans cet exemple 2D. La figure fait apparaître des zones blanches, sans points : il s'agit des zones mortes,

zones blanches : vu les résultats par identité et bijection ? Par la même zone parce que redondance.

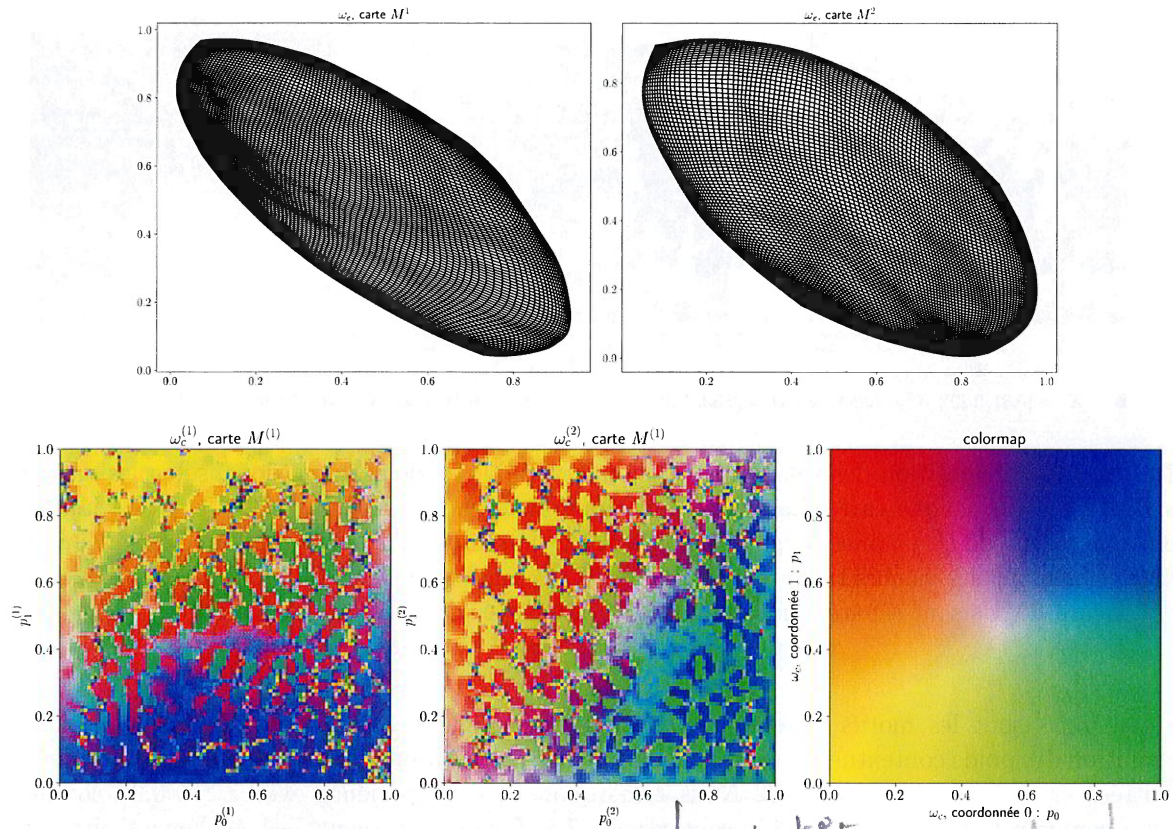


FIGURE 7.3 – Tracé des poids contextuels d'une architecture de cartes, organisées sur une sphère dans un espace 4D, avec  $r_e = 0.2$  et  $r_c = 0.02$ . Expliquez plus la figure.

séparant les zones de BMUs. Des BMUs proches codent pour des entrées ayant la même valeur de  $X$  et de  $U$ . Les points rouge et bleu sont au contraire envoyés dans des zones différentes, mais proches dans la carte. Contrairement aux cartes 1D, il est difficile de voir sur les tracés si  $U$  est une fonction de  $\Pi$  dans chaque carte. Le ratio de corrélation est calculable sur des valeurs en deux dimensions. Nous obtenons  $\eta(U|\Pi^{(1)}) = 0.99$  et  $\eta(U|\Pi^{(2)}) = 0.99$ . Ces valeurs sont de  $\eta(U|\Pi^{(1)}) = 0.94$  et  $\eta(U|\Pi^{(2)}) = 0.98$  pour des cartes ayant appris indépendamment sur les mêmes entrées 2D. En effet, la répartition des valeurs de  $U$  selon  $X^{(1)}$  et  $X^{(2)}$  donne déjà des ratios de corrélation élevés :  $\eta(U|X^{(1)}) = 0.91$  et  $\eta(U|X^{(2)}) = 0.98$ . Cet indicateur permet bien d'observer un apprentissage de  $U$  par le modèle CxSOM lorsqu'on le compare à la valeur des entrées. Une meilleure analyse de l'indicateur reste à étudier pour une étude claire de données en plus grande dimension, dans laquelle un tracé n'est plus possible.

### 7.2.2 Convergence des poids et motifs de poids contextuels

La formation de zones de poids contextuels dépend des paramètres de la carte et notamment rapport entre rayon de voisinage. Nous comparons dans cette section la forme des motifs de poids contextuels pour une même taille de  $r_e$  mais des rayons de poids contextuels  $r_c$  différents. Nous nous demanderons si les poids contextuels et externes convergent dans cette disposition et dans

Voisinage ray. de poids externes



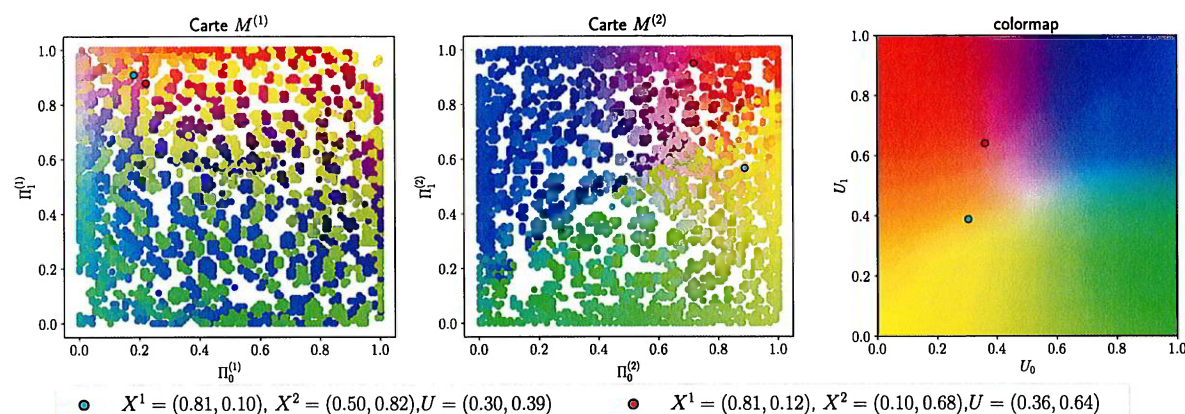


FIGURE 7.4 – Disposition des valeurs de  $U$ , ici une variable en deux dimensions, selon les valeurs du BMU II dans chaque carte. Les tracés font apparaître que des zones de BMUs se spécialisent pour une valeur de  $U$ , comme ce qui était observé en une dimension. Les points mis en valeurs en rouge et bleu ont des valeurs très proches d'entrée  $X^{(1)}$ , mais des valeurs différentes pour  $U$  et donc  $X^{(2)}$ . Leurs BMUs sont séparés sur la carte  $M^{(1)}$  et envoyés dans des zones distinctes.

ce cas, quels sont les motifs formés par les poids. Les Figures 7.7, 7.6 et 7.8 présentent ainsi l'évolution des poids contextuels au cours de l'apprentissage pour une même distribution de données d'entrées et  $r_c = 0.05, 0.03$  et  $0.02$ . Nous remarquons que l'expérience avec  $r_c = 0.02$  conduit à une configuration stable des poids contextuels, (7.8). Cette convergence est également observée dans l'expérience avec  $r_c = 0.03$  en figure 7.7. Au contraire, les poids contextuels ne montrent pas clairement de convergence pour  $r_c = 0.05$  : à l'issue des 500 000 itérations, nous pouvons remarquer des changements clairs dans la structure des poids contextuels. Les poids contextuels ne forment par ailleurs pas de motifs clairement définis, au contraire des deux jeux de paramètres précédents.

Cependant, la notion d'organisation n'est pas à évaluer : nous observons une organisation des poids contextuels similaires dans ces trois expériences. Les poids contextuels tendent en effet à s'organiser sous une forme hélicoïdale dans les trois expériences. Pour les cas où  $r_c = 0.02$  et  $r_c = 0.03$ , cette hélice est stable à partir de 200 000 itérations, mis à part quelques zones oscillant au cours des itérations. Dans le cas où  $r_c = 0.05$ , cette hélice évolue en tournant sur elle-même et apparaît toujours évoluer après 500 000 itérations. La similarité existant entre la forme des poids montre que les motifs dépendent bien de la relation entre entrées, ici la sphère.

Nous pouvons conclure de ces observations préliminaires en deux dimensions que la formation de zones est un mécanisme qu'on retrouve dans des cartes 1D comme des cartes 2D. Cependant, les deux dimensions apportent beaucoup moins de contraintes sur la forme des zones que sur des cartes en une dimension, rendant possible la formation différents motifs sur les poids contextuels. Contrairement au cas en une dimension dans lequel la convergence des poids était assurée, la convergence des poids contextuels dans une carte 2D n'est pas assurée en fonction des paramètres de la carte. Cette étude paramétrique en deux dimensions apparaît comme une suite nécessaire de l'étude des cartes. En Figure 7.9, nous avons tenté de représenter l'évolution des poids au cours de l'apprentissage, en traçant, tous les 5000 itérations, la moyenne des différences entre les poids des cartes au temps courant  $t$  et à la dernière configuration observée ( $t - 5000$ ). Ce tracé ne constitue pas une preuve expérimentale de convergence, mais nous donne une idée de l'évolution des poids également observée à l'œil. Pour  $r_c = 0.02$ , nous relevons une différence moyenne entre les poids au

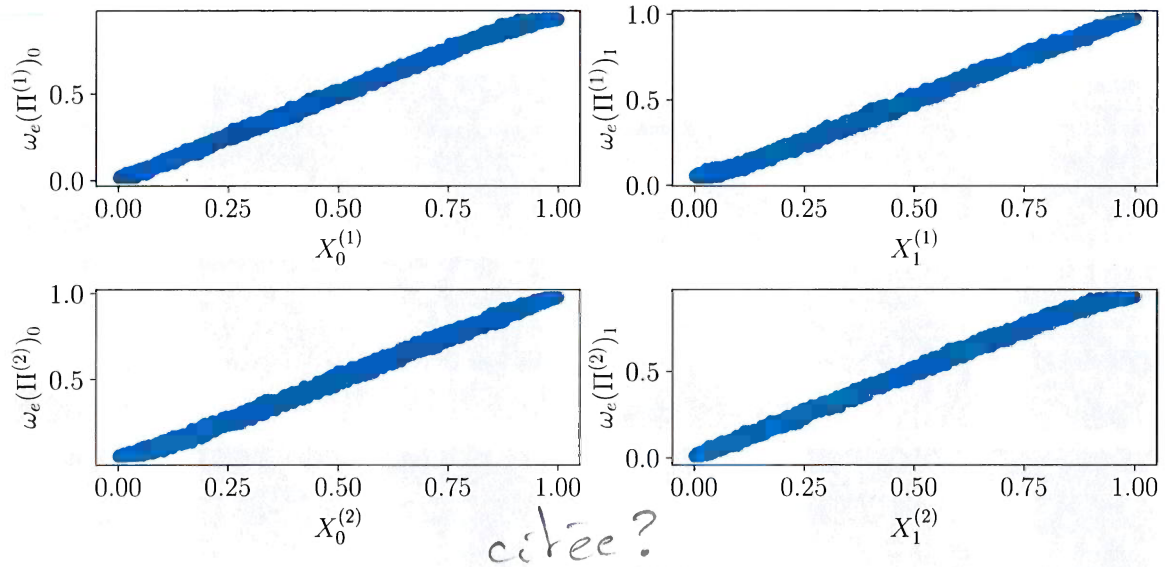


FIGURE 7.5 – Tracé de l'erreur de quantification vectorielle des poids externes sur l'architecture de deux cartes. Les coordonnées de chaque élément sont tracées séparément.

temps  $t$  de 0.01 à l'itération 500000. Plusieurs travaux dans la littérature proposent des outils et des critères pour étudier la convergence et l'organisation de cartes de Kohonen (Polani 2001; Breard 2017; Tatoian et Hamel 2018), qu'il sera intéressant d'utiliser également dans les architectures CxSOM.

Nous avons analysé la forme des poids sur un seul type de distribution, une sphère dans un espace 4D, pour différents rayons de voisinage de la carte. Nous observons que les motifs formés par les poids contextuels sont similaires dans tous ces cas et ne dépendent donc pas des conditions initiales des expériences : les poids des cartes sont initialisés aléatoirement de façon différente dans chaque expérience et les entrées sont tirées sur une même distribution de façon aléatoire. Nous pouvons donc supposer que le type de relation entre entrées influence l'organisation des motifs des poids contextuels. L'étude des comportements d'une carte 2D en fonction du type entrées est une suite pertinente de ces résultats préliminaires.

### 7.2.3 Convergence de la relaxation

Nous nous intéressons ensuite à la convergence de la relaxation sur l'expérience en deux dimensions. Nous avons montré qu'un BMU a du sens dans une carte si la relaxation converge vers un point fixe. Nous traçons en figure l'évolution du taux de convergence des tests au cours de l'apprentissage pour des structures de deux cartes en deux dimensions, par le processus décrit au chapitre (??). Lors de plusieurs phases de test lancées à des itérations  $t$  au cours de l'apprentissage, nous comptons le nombre de pas de relaxation nécessaires à la convergence de la recherche de BMU et traçons sur la figure la moyenne de ce nombre de pas de relaxation à chaque phase  $t$ . Nous traçons également sur la deuxième figure le taux d'échantillons d'un phase de test menant à une convergence de la relaxation. On considère que la relaxation n'a pas convergé si le nombre de pas atteint le seuil maximal de 1000 pas fixé par notre étude. Le nombre de pas de relaxation moyen étant observé de 20 pas, ce seuil est pertinent pour considérer que la relaxation n'a pas convergé.



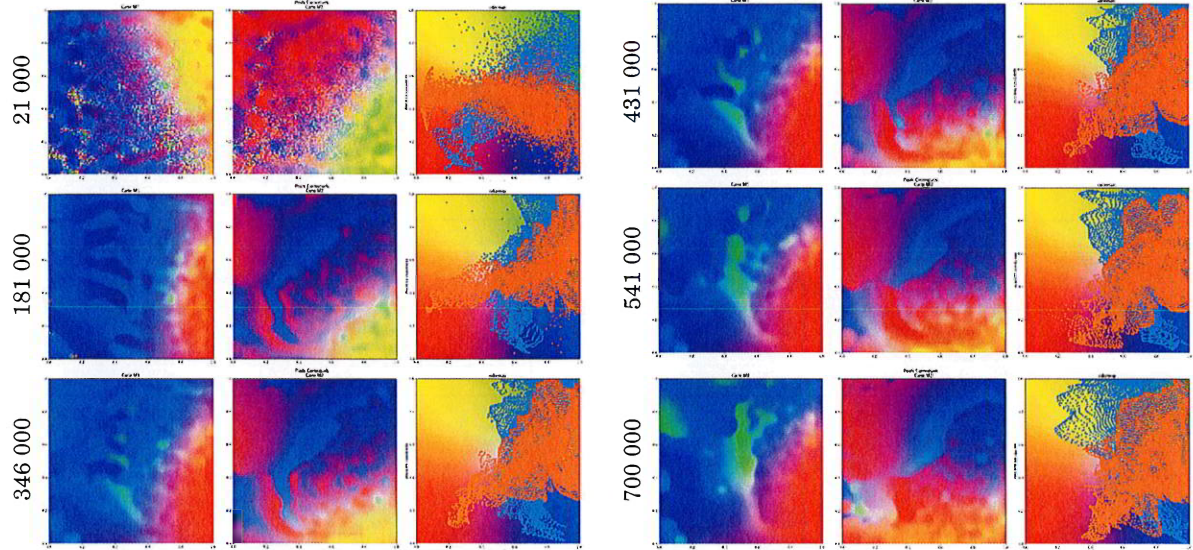


FIGURE 7.6 – Évolution des poids contextuels d’une architecture de deux cartes pour  $r_c = 0.05$ . Les points sont représentés sous forme de carte de coloration à gauche et sous forme de disposition dans l’espace des positions sur la figure de droite. On observe une évolution régulière sous forme de motif en hélice, mais cette disposition évolue dans le temps. Après 700000 itérations, la figure semble se déformer vers de nouveaux motifs. Cette organisation ne converge pas. Les valeurs prises par les poids contextuels des cartes  $M^{(1)}$  (bleu) et  $M^{(2)}$  (orange) sont tracés sur la carte de coloration, faisant apparaître un motif hélicoïdal évoluant dans le temps.

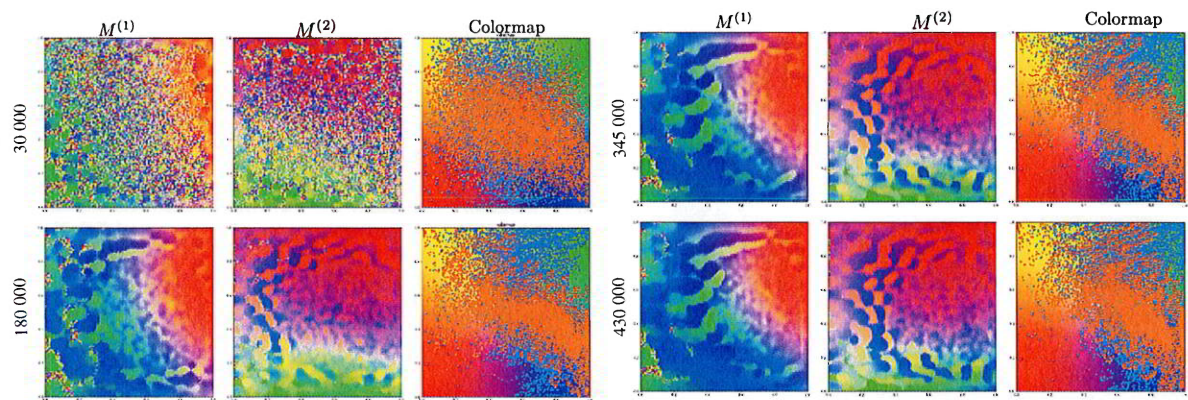


FIGURE 7.7 – Évolution des poids contextuels d’une architecture de deux cartes pour  $r_c = 0.03$ . Les motifs observés sont similaires à ceux observés pour d’autres paramètres. Les poids présentent toujours une forme hélicoïdale évoluant dans le temps.



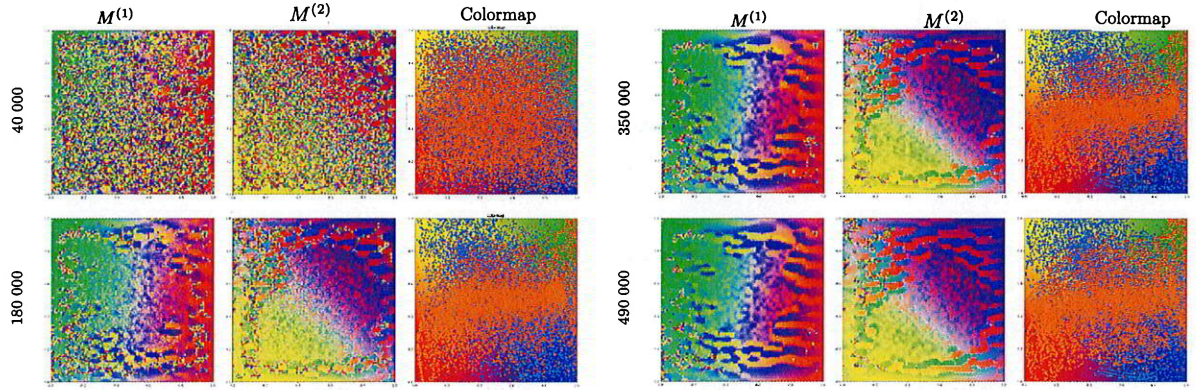


FIGURE 7.8 – Évolution des poids contextuels d’une architecture de deux cartes pour  $r_c = 0.02$ . Les motifs observés sont similaires à ceux observés pour d’autres paramètres. Les poids présentent toujours une forme hélicoïdale évoluant dans le temps.

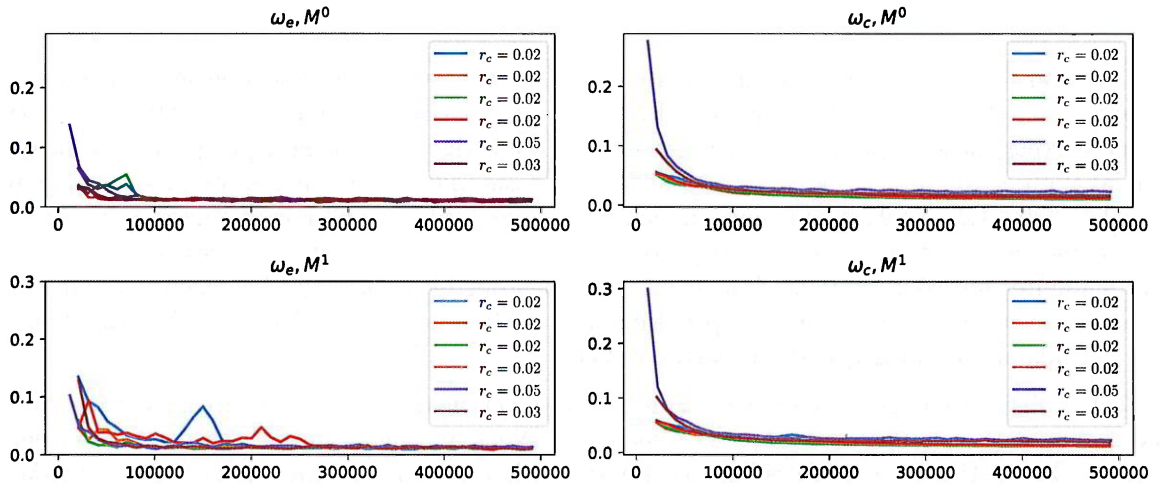


FIGURE 7.9 – Évolution de la différence moyenne entre les poids externes et contextuels au cours de l’apprentissage d’une architecture de 2 cartes sur une sphère 3D en 4 dimensions. Le calcul de différences est réalisé toutes les 10 000 itérations. Nous avons tracé les courbes pour 4 expériences avec  $r_e = 0.2$  et  $r_c = 0.02$  ainsi que pour les expériences avec  $r_e = 0.2$  et  $r_c = 0.05$ ,  $r_e = 0.2$  et  $r_c = 0.03$ . Nous remarquons que cette différence évolue vers une valeur faible, qu’on attend nulle pour une carte stabilisée. Ici les poids tendent vers une valeur faible qui semble stable. Notons que cela ne montre pas forcément la convergence : les poids peuvent se déplacer faiblement dans une même direction. Le tracé nous assure simplement que l’évolution des poids est lente en fin d’apprentissage.

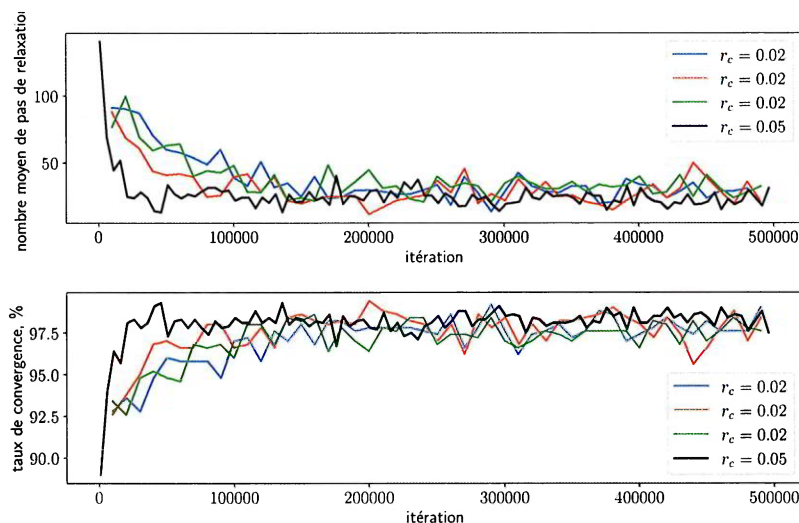


FIGURE 7.10 – Évolution de la convergence de la relaxation au cours de l'apprentissage. Nous avons réalisé les tracés sur trois expériences générées pour  $r_c = 0.02$ , sur des entrées aléatoires tirées sur la même distribution, une sphère 4D. Pour comparaison, nous traçons également l'évolution pour l'expérience avec  $r_c = 0.05$ , dans laquelle les poids contextuels n'ont pas convergé. La relaxation converge en fin d'apprentissage : le BMU a donc un sens dans les cartes en deux dimensions.

Nous traçons ces évolutions pour différents paramètres de cartes : trois expériences de mêmes paramètres  $r_e = 0.2$  et  $r_c = 0.02$  ainsi que les expériences avec  $r_c = 0.03$  et  $r_c = 0.05$ . Dans cette dernière expérience, les poids n'ont pas formé de zones distinctes (voir figure ??). Nous voyons que la relaxation converge bien dans 95 et 98 % des cas tout au long de l'apprentissage. Comme dans les cartes 1D, la relaxation converge peu lorsque les poids ne sont pas organisés et le taux de convergence augmente au cours de l'organisation des cartes. Les valeurs trouvées pour  $r_c = 0.05$  sont similaires au cas  $r_c = 0.02$ , dans lequel les poids ont bien convergé. La relaxation dans des cartes en deux dimensions n'est donc pas liée à la formation de zones stables : la relaxation trouve un point fixe dans un cas général de cartes en 2D.

Cette observation est prometteuse pour la construction d'architectures. Cela montre que le BMU a un sens dans une carte en deux dimensions. Nous n'avons pas étudié en deux dimensions l'unicité du BMU en fonction des valeurs d'initialisation de la relaxation. Nous avons cependant observé que la relaxation converge la plupart du temps, et qu'elle converge vers une valeur dont le poids  $\omega_e(\Pi)$  est proche de l'entrée externe d'après la faible erreur de quantification vectorielle observée sur les entrées externes en figure 7.5.

#### 7.2.4 Prédiction d'entrée

Nous avons vu que les cartes en deux dimensions s'organisent, comme en 1D, de manière à former des zones dans les poids contextuels. Nous avons également observé que la recherche de BMU a un sens, car la relaxation converge dans les cartes 2D. Nous nous attendons donc à ce qu'une architecture de cartes 2D soit en mesure de générer une prédiction.

Pour cela, nous construisons une architecture de trois cartes en deux dimensions. Chaque carte



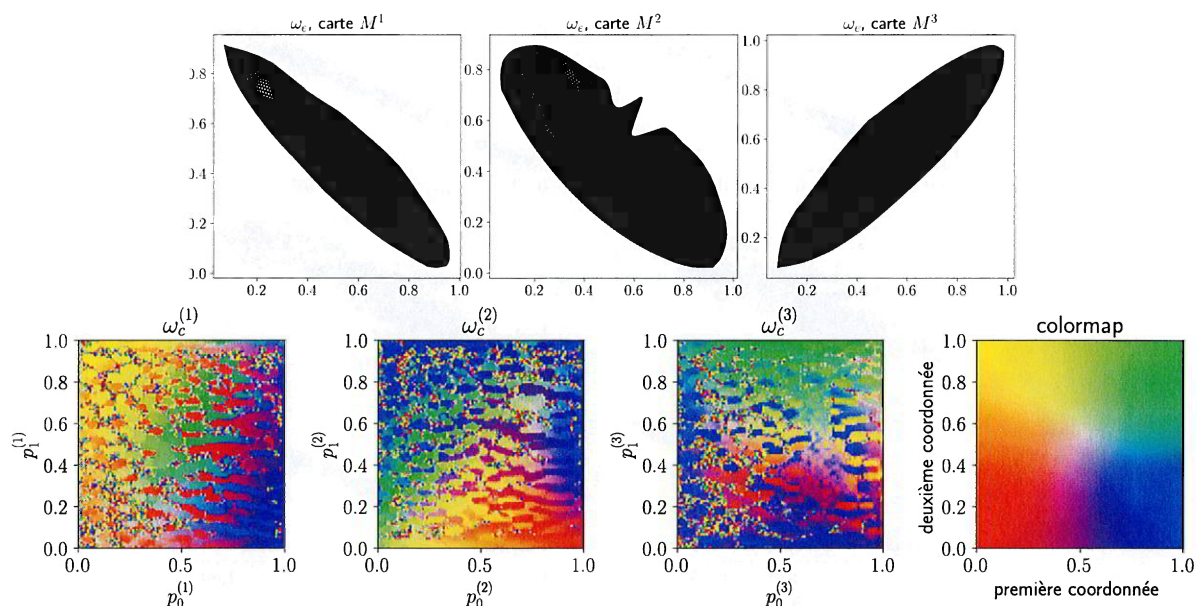


FIGURE 7.11 – Tracés des poids externes (en haut) dans l’espace des entrées et des poids contextuels (en bas) sous forme de carte de coloration. Les motifs formés par les poids contextuels sont similaires à ceux observés sur deux cartes.

prend en entrée externe une paire de coordonnées d’un espace multimodal en 6D. Ces entrées sont situées sur une sphère de dimension 3 plongée dans l’espace en 6D par le même processus de rotation qu’en 4D.  $U$  est donc toujours une variable 2D. Dans cette configuration, la connaissance de deux entrées sur trois et du modèle détermine la troisième : la prédiction est donc possible. Nous prenons  $r_e = 0.2$  et  $r_c = 0.02$ , pour des cartes de taille  $100 \times 100$ . Les cartes sont entraînées sur 250 000 itérations, à l’issue desquelles les poids externes contextuels ont atteint une position stable.

Nous traçons d’abord les poids externes et contextuels des trois cartes sont représentés en Figure 7.11. L’organisation des poids contextuels conduit à la présence de motifs très semblables à ceux observés sur une architecture de deux cartes.

Nous lançons une étape de prédiction à la fin de l’apprentissage lors de laquelle la carte  $M^{(2)}$  ne reçoit plus d’entrée externe. La valeur de  $\omega_e^{(2)}(\Pi^{(2)})$  est alors utilisée comme prédiction de l’entrée  $X^{(2)}$ . La Figure 7.12 indique la valeur prédite en fonction de l’entrée  $X^{(2)}$ . Les valeurs de  $X$  et  $\omega_e$  étant 2D, nous avons représenté séparément chaque dimension. Cette expérience montre que la prédiction est correctement réalisée, donc que l’architecture a appris le modèle des entrées 2D. L’erreur de prédiction observée dans cet exemple est cependant très élevée par rapport à l’échelle de quantification vectorielle de la carte. Ce comportement sera donc à évaluer sur d’autres distributions d’entrée.

### 7.2.5 Organisation de cartes sur des entrées indépendantes

Nous nous intéressons enfin à l’organisation d’une architecture de deux cartes prenant des entrées  $X^{(1)}, X^{(2)} \in [0, 1]^2 \times [0, 1]^2$  indépendantes. Nous avons également observé la présence de zones distinctes de BMUs lorsque deux cartes apprennent sur des entrées dépendantes. Nous nous plaçons ici dans un cas limite et cherchons à observer si les zones de BMUs sont encore présentes.

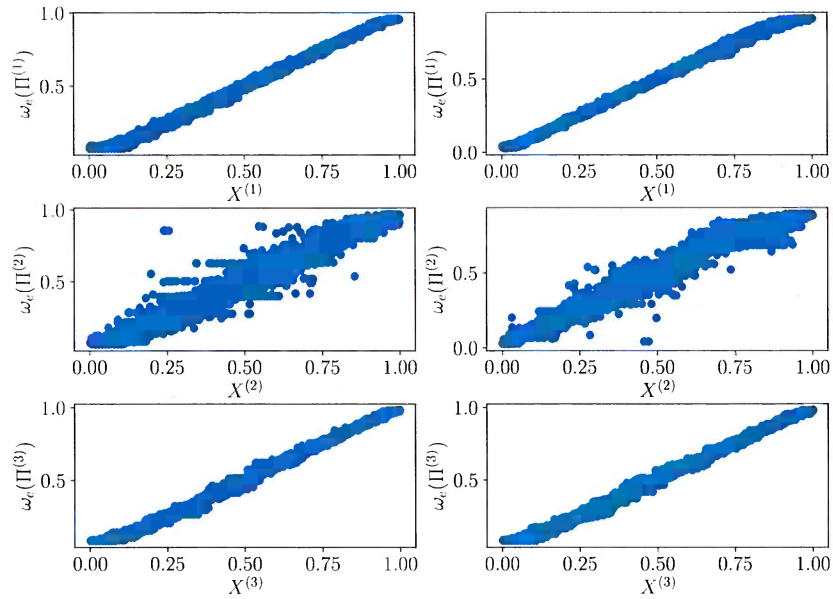


FIGURE 7.12 – Tracé de l'erreur de prédiction  $\omega_e^{(2)}(\Pi^{(2)})$  en fonction de la valeur théorique de  $X^{(2)}$ , non présentée à l'architecture, dans une architecture de trois cartes 2D prenant des entrées  $X^{(i)}$  en deux dimensions  $[X_0^{(i)}, X_1^{(i)}]$ . Nous traçons sur une ligne, pour chaque entrée, les dépendances entre chacune des dimensions. Lorsque la carte  $M^{(2)}$  ne reçoit pas d'entrée externe. Les cartes  $M^{(1)}$  et  $M^{(3)}$  ayant une activité externe, le graphique montre que la quantification vectorielle est bien réalisée dans ces cartes. La carte  $M^{(2)}$  est uniquement activée par les connexions contextuelles venant de  $M^{(1)}$  et  $M^{(3)}$ . La figure du haut montre que la prédiction est correctement réalisée, en observant cependant une erreur très forte.

Nous prenons des rayons de voisinage  $r_e = 0.2, r_c = 0.05$ .

Nous nous intéressons ici aux motifs formés par les poids externes et contextuels sur chaque carte. Les valeurs de ces poids externes et contextuels sont tracés en figure 7.13. Tracer les poids nous permet ici de comparer l'organisation d'une carte à celle observée en une dimension. Nous soulignons cependant que cette représentation ne permet pas de détecter quelles unités sont effectivement BMU lors d'un test.

Les poids externes sont bien dépliés sur l'ensemble des entrées, ce qui est similaire au cas en une dimension. Les poids contextuels restent centrés autour de 0.5 : nous faisons apparaître en bleu et orange sur la carte de coloration les valeurs effectivement prises par les poids contextuels de chaque carte. Pourtant, les positions des BMUs de chaque carte observées lors d'un test s'étendent bien sur toute la surface d'une carte. Les poids contextuels auraient dû, pour bien cartographier les positions sur l'autre carte, s'étendre sur tout le carré dans chaque zone. Ce comportement est à rapprocher du comportement limite observé sur une architecture de 10 cartes en une dimension : nous avons vu que des architectures de 10 cartes apprenant sur 10 entrées 1D indépendantes voient également leurs poids contextuels se moyenniser autour de 0.5 dans chaque carte. Proposons une interprétation de ce comportement : lorsque les entrées présentent une dépendance, un nombre fini ou un intervalle réduit de valeurs de  $X^{(2)}$  seront présentées à l'architecture pour une même valeur de  $X^{(1)}$ , correspondant à autant d'entrées contextuelles pour  $M^{(1)}$ . Les poids contextuels situés autour de la position codant pour  $X^{(1)}$  s'organisent alors comme une carte des valeurs des



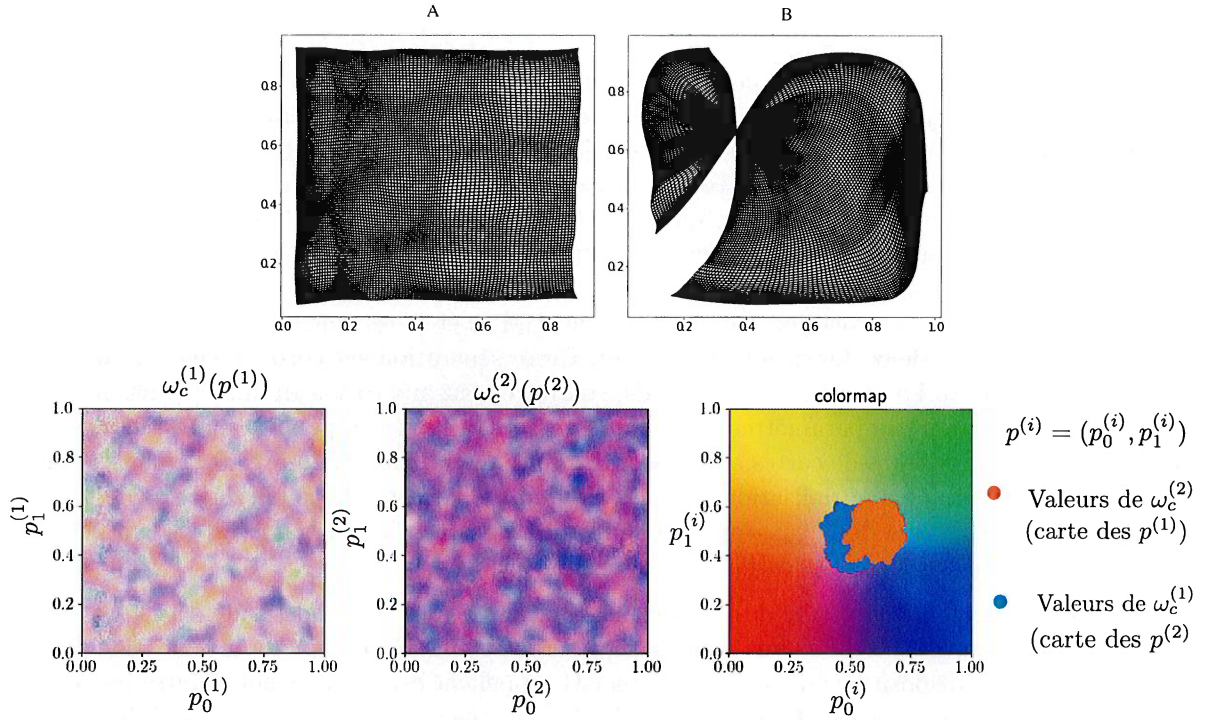


FIGURE 7.13 – En haut : poids externes des cartes  $M^{(1)}$  et  $M^{(2)}$  représentés sous forme de distorsion de la carte après 200000 itérations. En bas : poids contextuels des cartes pour la même itération, représentés sous forme de carte de couleur en deux dimensions. Un pixel situé à la position  $p_i, p_j$  prend comme couleur correspondante la valeur 2D de son poids contextuel, associé à une couleur par la carte de coloration représentée à droite de la figure. Les points orange et bleus indiqués sur la carte de coloration sont les valeurs effectivement prises par toutes les valeurs de  $\omega_c^{(1)}$  et  $\omega_c^{(2)}$ , cartographiant les positions des BMUs de l'autre carte. On remarque donc que les poids contextuels ne se déploient pas sur toutes les valeurs prises par les BMUS, car toutes les unités de la carte ont été BMU lors de cette phase de test.

entrées contextuelles présentées. En une dimension, cette sous carte avait possibilité de s'étendre sur tout l'intervalle de valeurs de  $X^{(2)}$ , formant donc des zones distinctes. En ajoutant des entrées contextuelles, les poids tendaient vers une valeur moyenne des positions, donc 0.5. En deux dimensions, pour une même valeur de  $X^{(1)}$ , toutes les valeurs possibles de  $X^{(2)}$  dans le carré  $[0, 1]^2$  sont présentées. Les poids contextuels n'ont pas la liberté de s'étendre sur toutes ces valeurs et prennent donc une valeur moyenne.

Ce comportement peut apparaître comme une limite de CxSOM car les poids contextuels manquent de liberté pour s'organiser. Au contraire, nous pouvons aussi envisager ce comportement comme une détection de dépendance entre entrées. Ce comportement à la limite de CxSOM est une piste de propriété à étudier pour des travaux futurs.

### 7.3 Conclusion

Les expériences proposées dans ce chapitre sur les cartes en deux dimensions donnent une idée des comportements auxquels on peut s'attendre dans un cas plus général d'architecture. Ces expériences nous ont également permis de vérifier la méthode d'étude de carte construite tout au long de cette thèse et constituent une première application des indicateurs proposés au chapitre précédent. Nous avons d'abord observé que l'émergence de zones de poids contextuels est une propriété qui se transpose sur des cartes en deux dimensions. Ces zones constituent donc un point clé des architectures CxSOM. La formation de ces zones permet, comme sur des cartes 1D, de séparer les valeurs de  $U$  en fonction des positions des BMUs et  $U$  est une fonction du BMU dans chacune des cartes en deux dimensions également. Cette séparation est correctement évaluée par le ratio de corrélation. La convergence des poids, contrairement aux cartes en une dimension, n'est pas assurée en fonction des paramètres d'apprentissage et constitue un point à analyser lors de la construction d'architectures de cartes en deux dimensions. Par ailleurs, nous avons déplié les poids externes lors d'une étape d'initialisation afin de s'assurer que la carte 2D ne présente pas de "point de torsion". La robustesse de l'algorithme CxSOM sur des cartes "mal organisées" est un point à étudier avant une éventuelle application de l'algorithme.

Nous remarquons que la forme des motifs spatiaux formés par les poids contextuels sont plus variés que dans le cas des cartes en une dimension ; il serait intéressant d'explorer la formation de ces motifs sur d'autres dispositions d'entrées. Ces motifs rappellent ceux émergeant d'autre processus d'organisation, tels que les motifs de Turing. Ces motifs émergent de processus de diffusion, ce qui n'est a priori pas le cas sur les cartes de Kohonen : l'organisation des cartes est guidée par la présentation d'une séquence aléatoire d'entrées. Il est cependant intéressant d'observer la similarité existant dans les motifs, qui montrent que l'organisation des cartes est issue d'une dynamique complexe (au sens des systèmes complexes). Les poids contextuels montrent également l'émergence de motifs temporels lors de l'apprentissage : l'organisation en hélice pour  $r_c = 0.05$ . Ces observations montrent que les mécanismes d'apprentissage émergeant dans des cartes en deux dimensions sont plus complexes qu'en une dimension. Cela peut être un atout pour le modèle car cela laisse la porte ouverte à des comportements plus complexes sur des cartes 2D ; mais cela apporte plus de difficulté pour une éventuelle compréhension du modèle. Ainsi, il est pertinent d'étudier l'organisation des cartes d'un point de vue macroscopique pour la suite des travaux. Si la forme des motifs en 1D nous apportait une compréhension du comportement d'une architecture de trois cartes, il est compliqué de prédire l'apprentissage d'une architecture de cartes 2D et de plus nombreuses cartes à partir de la forme des poids.



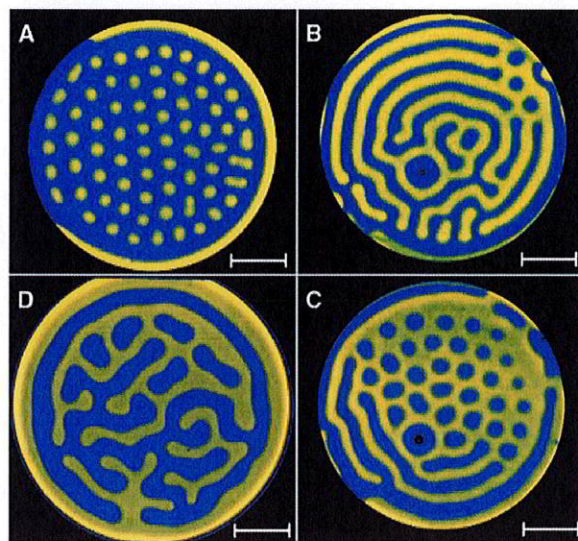


FIGURE 7.14 – Motifs de Turing observés lors d’une réaction de diffusion chimique. Ces motifs émergent après l’introduction d’une perturbation dans un système initialement stable. Les différents motifs A,B,C,D sont des états atteints dans différentes expériences en fonction des paramètres de la réaction. Dans nos cartes, nous observons des motifs similaires au motif A lorsque  $r_c = 0.02$ , mais nous observons également des motifs rappelant C et D pour des rayons de voisinage différents. Notons cependant que l’organisation des cartes n’est a priori pas un processus de diffusion et est générée par la présentation d’une séquence d’entrées.





# Bibliographie

- Breard, Gregory. "Evaluating Self-Organizing Map Quality Measures as Convergence Criteria". In : 2017 ( 7).
- Cottrell, Marie, Jean-Claude Fort et Gilles Pagès. "Theoretical aspects of the SOM algorithm". In : *Neurocomputing* 21 (1998), p. 119-138 ( 3).
- Flanagan, John A. "Self-organisation in Kohonen's SOM". In : *Neural Networks* 9.7 (1996), p. 1185-1197 ( 3).
- Polani, Daniel. "Measures for the organization of self-organizing maps". In : 2001 ( 7).
- Tatoian, Robert et Lutz Hamel. "Self-Organizing Map Convergence". In : *Int. J. Serv. Sci. Manag. Eng. Technol.* 9 (2018), p. 61-84 ( 7).

