

## Chapitre 3

# Analyse du mécanisme de relaxation

### Sommaire

<b>3.1 Introduction</b>	<b>65</b>
3.1.1 Du calcul cellulaire au calcul à l'échelle d'une carte	65
3.1.2 Problématique du chapitre	67
<b>3.2 Formalisme de l'algorithme de relaxation</b>	<b>67</b>
3.2.1 Formulation de l'évolution des BMUs lors de la relaxation	68
3.2.2 Formulation du problème d'optimisation	69
3.2.3 Relaxation et notion de <i>Best Matching Unit</i>	70
<b>3.3 Étude expérimentale de la convergence de la relaxation</b>	<b>71</b>
<b>3.4 Étude de l'évolution de la relaxation</b>	<b>73</b>
3.4.1 Trajectoires de relaxation	73
3.4.2 Influence du pas de relaxation	76
<b>3.5 Conclusion</b>	<b>79</b>

### 3.1 Introduction

#### 3.1.1 Du calcul cellulaire au calcul à l'échelle d'une carte

Le processus de relaxation que nous avons défini au chapitre précédent dans l'algorithme CxSOM est une méthode originale pour construire des connexions bidirectionnelles entre cartes. Dans ce chapitre, nous chercherons à observer le comportement de la relaxation et notamment sa convergence dans les cartes de l'architecture CxSOM. Avant cela, nous nous intéressons à son inspiration biologique et pourquoi il reste proche des architectures de cartes cellulaires développées précédemment dans l'équipe (Bassem Khouzam 2014; Ménard et Frezza-Buet 2005), qui ont inspiré ces travaux.

Dans ces modèles d'architectures, le processus dynamique d'apprentissage s'appuyait sur des champs neuronaux dynamiques couplés (DNF), qui remplaçaient la recherche de BMU. Les DNF, introduits en (Amari 1977) permettent de modéliser à gros grains l'activité spatiotemporelle d'une population de neurones impulsifs. Au lieu de s'intéresser à l'activité d'un seul neurone, tel que son taux de décharge, le modèle des DNF calcule un potentiel  $u(x, t)$  en un ensemble de positions continues  $x$  à un temps  $t$ , réagissant à un stimulus d'entrée  $s(x, t)$ . Les positions  $x$  s'étendent sur l'espace des caractéristiques d'entrée et représentent une valeur dans cet espace. Ainsi, un DNF réagissant à un stimulus 1D est en une dimension, etc. L'activation du DNF est ensuite une approximation du taux de décharge moyen du champ de neurones à chaque position  $x$ .  $u(x, t)$  s'exprime à partir d'une équation différentielle ; son évolution prend en compte le potentiel  $u(y, t)$  aux positions voisines de  $x$ , pondérées par un terme d'interaction latérale. Ce terme est en général une différence de gaussiennes, rendant le voisinage proche de  $x$  excitateur : son potentiel renforce le potentiel  $u(x, t)$ , tandis que le voisinage à plus longue distance est inhibiteur, amenant une dynamique de *Winner Take All*. Le calcul du potentiel des DNF sur un stimulus d'entrée résulte en la formation de bulles d'activités dans les zones où le signal d'entrée est élevé.

Les DNF ont plusieurs applications possibles en fonction de leur paramétrage, comme le filtrage de bruit ou le suivi d'une cible positionnée en  $x$ . Comme le calcul du potentiel s'appuie uniquement sur un voisinage de  $x$  en chaque point, les calculs permettant ce mécanisme Winner

Take All sont cellulaires. À partir de cela, des DNF réagissant à plusieurs signaux d'entrée  $s_i(x)$  peuvent être couplés en des architectures dynamiques exhibant des mécanismes complexes (J. D.

Fix et al. 2011 ; Sandamirskaya 2014). Le couplage de DNF permet d'envisager des comportements autonomes au sein des structures de DNF et non une seule réaction à l'entrée  $s$ . La construction d'architectures de cartes auto-organisatrices cellulaires en (Bassem Khouzam 2014 ; Ménard et Frezza-Buet 2005) utilise des DNF couplés entre cartes afin de choisir, de façon cellulaire, le BMU d'une carte, c'est à dire l'unité dont le poids va être mis à jour. Les architectures de cartes proposées dans ces travaux apportent alors un mécanisme d'apprentissage aux DNF qui sont eux-mêmes seulement un processus dynamique réagissant aux entrées. Les DNF apportent quant à eux le calcul cellulaire manquant aux cartes auto-organisatrices classiques pour une implémentation au niveau du neurone.

La recherche de BMU d'une carte de Kohonen implémente également un mécanisme de Winner Take All, [mais] dont le calcul est centralisé à l'échelle de la carte grâce au calcul de l'activation des neurones. Ce calcul centralisé nécessite moins de ressources de calcul que le calcul des DNF, qui s'appuie sur la résolution numérique d'une équation différentielle.

Dans le même esprit que l'utilisation d'une activation chez Kohonen pour trouver le BMU au lieu d'un mécanisme cellulaire, l'algorithme de relaxation que nous avons proposé cherche à conserver un processus dynamique de recherche de BMU mais en plaçant les calculs à l'échelle d'une carte. Le choix d'un BMU II dans une carte auto-organisatrice par un  $\arg \max$  équivaut

à la génération d'une bulle d'activité centrée en II dans un DNF. Le processus dynamique de relaxation que nous utilisons dans le modèle CxSOM se rapproche ainsi des mécanismes réalisés dans les DNF couplés, mais en remplaçant le processus cellulaire réalisé dans une carte par le calcul d'activité et le choix du BMU. Le couplage entre les activités des cartes reste par contre un processus dynamique. Ce choix nous a permis de simplifier les calculs réalisés au sein d'une carte afin d'envisager d'étudier facilement des grandes architectures de cartes, tout en conservant une proximité avec une version cellulaire des cartes auto-organisatrices.

Dans notre modèle, le mécanisme de relaxation cherche à la fois à trouver un BMU dans chaque carte, c'est à dire la position à laquelle l'activité est maximale, et joue le rôle d'interface entre cartes en utilisant la position du BMU dans les cartes voisines pour le calcul d'activité.

↳ Tu cherches à faire le lien entre relax et DNF pour répondre à une inspiration bio-donc ?

#### 3.1.2 Problématique du chapitre

Dans ce chapitre, nous nous intéressons à la relaxation en tant que mécanisme d'interface entre cartes. Nous cherchons d'abord à décrire formellement l'algorithme de relaxation proposé dans le modèle CxSOM afin de pouvoir mieux analyser et représenter sa dynamique. Nous nous demanderons notamment si la relaxation est équivalente, comme dans une carte simple, à une recherche de maximum d'activation et expliciterons les propriétés de convergence qu'on attend de l'algorithme de relaxation pour pouvoir le considérer comme une recherche de BMU. Nous attendons notamment que la recherche de BMU converge et que le point de convergence soit unique. Le comportement de la relaxation dépend complètement des configurations des poids externes et contextuels des cartes de l'architecture. Ces configurations de poids résultent d'un processus de mise à jour et n'ont pas de formulation analytique. Ce chapitre cherchera à observer l'évolution de la relaxation dans différentes configurations de poids des cartes et de paramètres afin de répondre aux questions de convergence que nous aurons formulées.

↳ d'apprentissage ?

### 3.2 Formalisme de l'algorithme de relaxation

La recherche du BMU par relaxation dans l'architecture CxSOM se traduit par une recherche du maximum des activations de chacune des cartes de l'architecture. La relaxation est une heuristique de recherche de ce maximum. Nous décrivons dans cette section la relaxation sous une forme plus mathématique et formulons le problème d'optimisation qu'elle cherche à résoudre. Nous introduisons ainsi des notations plus détaillées qui nous permettront d'observer les quantités relatives à l'évolution de la relaxation en section suivante.

### 3.2.1 Formulation de l'évolution des BMUs lors de la relaxation

La relaxation définie dans le modèle est une recherche de maximum par déplacements dans l'espace des positions de chaque carte  $(p^{(1)}, \dots, p^{(n)})$ . Nous notons  $(\mathbf{\Pi})_\tau = (\Pi_\tau^{(0)}, \dots, \Pi_\tau^{(n)})$  la suite définie par l'évolution des BMUs des  $n$  cartes d'une architecture lors de la relaxation. Détaillons ici l'évolution de cette suite. ) ? de ?

Nous définissons également pour chaque carte  $i$ , une seconde suite  $\hat{p}_\tau^{(i)}$ , correspondant à la position maximisant l'activité globale de la carte  $i$  à l'instant  $\tau$  :

$$\hat{p}_\tau^{(i)} = \arg \max_p (a_g^{(i)}(p, \Pi_\tau^{(i_0)}, \dots, \Pi_\tau^{(i_K)})) \quad (3.1)$$

$i_0, \dots, i_K$  indices des cartes nourrissant la carte  $i$ .

L'équation d'évolution de la suite  $(\mathbf{\Pi})_\tau$  s'écrit à partir des valeurs  $\hat{p}_\tau$  :

$$\Pi_{\tau+1}^{(i)} = \begin{cases} \Pi_\tau^{(i)} + \text{sgn}(\hat{p}_\tau^{(i)} - \Pi_\tau^{(i)}) \times \Delta & \text{si } \hat{p}_\tau^{(i)} - \Pi_\tau^{(i)} > \Delta \\ \hat{p}_\tau^{(i)} & \text{sinon} \end{cases} \quad (3.2)$$

$a_g^{(i)}$  est une fonction des poids de la carte  $\omega_e^{(i)}, \omega_c^{(i)}$  et de son entrée externe  $X^{(i)}$ . Lors du processus de relaxation, les poids et l'entrée  $X^{(i)}$  restent fixes. Le calcul de  $a_g$  à l'instant  $\tau$  ne dépend donc pas de  $\tau$ . Pour toute carte  $i$ ,  $\hat{p}_\tau^{(i)}$  dépend donc uniquement de  $(\Pi_\tau^{(i_0)}, \dots, \Pi_\tau^{(i_K)})$ . En posant  $f^{(i)}$  toute la partie droite de l'équation 3.2, on peut finalement écrire :

$$\forall i, \Pi_{\tau+1}^{(i)} = f^{(i)}(\Pi_\tau^{(0)}, \dots, \Pi_\tau^{(n)}) \quad (3.3)$$

Soit, pour l'ensemble des composantes :

$$\mathbf{\Pi}_{\tau+1} = \mathbf{f}(\mathbf{\Pi}_\tau) \quad (3.4)$$

Si  $(\mathbf{\Pi})_\tau$  converge, alors elle converge vers un point fixe de la fonction  $\mathbf{f}$ , soit une position  $\mathbf{\Pi}$  vérifiant :

$$\mathbf{\Pi} = \mathbf{f}(\mathbf{\Pi}) \quad (3.5)$$

Cependant, rien ne garantit que des points fixes existent ni que la suite converge. Pour des poids  $\omega$  quelconques, il n'existe généralement pas de point fixe.

Notons enfin que l'évolution de la suite  $(\mathbf{\Pi})_\tau$  dépend de son initialisation. Lors de l'appren-

tissage du modèle, nous prenons comme état initial une position  $(\Pi_0^{(0)}, \dots, \Pi_0^{(n)})$  telle que :

$$\begin{cases} \Pi_0^{(0)} = \arg \max_{p^{(0)}} a_e(p^{(0)}, X^{(0)}) \\ \dots \\ \Pi_0^{(n)} = \arg \max_{p^{(n)}} a_e(p^{(n)}, X^{(n)}) \end{cases} \quad (3.6)$$

Nous étudierons dans ce chapitre la relaxation pour  $(\Pi_0^{(0)}, \dots, \Pi_0^{(n)})$  quelconques.

### 3.2.2 Formulation du problème d'optimisation

La relaxation que nous venons de décrire est la recherche d'un ensemble de valeurs  $\mathbf{\Pi} = (\Pi^{(1)}, \dots, \Pi^{(n)})$ .

Nous pouvons interpréter la relaxation comme la recherche d'un maximum de l'activité totale de l'architecture de cartes, que nous détaillons ici.

Rappelons les équations de calcul d'activation : Dans chaque carte  $i$ , l'activité globale est définie par :

$$a_g^{(i)}(p, X, \gamma_0, \dots, \gamma_K) = \sqrt{a_e(p, X) \left( \frac{1}{2} a_e(p, X) + \frac{1}{2} a_c(p, \gamma_0, \dots, \gamma_K) \right)} \quad (3.7)$$

Avec  $\gamma_0, \dots, \gamma_K$  les  $K$  entrées contextuelles de la carte.

L'activité contextuelle  $a_c$  est définie, rappelons le, comme la moyenne des activités contextuelles sur chaque couche de poids contextuels :

$$a_c(p, \gamma_0, \dots, \gamma_K) = \frac{1}{K+1} \sum_{k=0}^K a_{ck}(p, \gamma_k) \quad (3.8)$$

Pour simplifier les notations, nous noterons que les entrées contextuelles d'une carte  $i$  sont  $\gamma_k, k = 0 \dots n, k \neq i$ .

$a_g^{(i)}$  est à valeurs positives pour tout  $i$  ; maximiser individuellement  $a_g^{(i)}$  revient à maximiser la somme des  $a_g$ . La recherche de BMU s'exprime comme une solution  $(\Pi^{(0)}, \dots, \Pi^{(n)})$  du problème d'optimisation suivant :

$$\begin{cases} \text{Maximiser} & \sum_{i=1}^n a_g^{(i)}((\Pi^{(i)}, X^{(i)}, \gamma_0^{(i)}, \dots, \gamma_n^{(i)})) \\ \text{Sous Contrainte} & \forall i, \forall k \neq i, \gamma_k^{(i)} = \arg \max_p a_g^{(k)}(p, \gamma_0^{(k)}, \dots, \gamma_n^{(k)}) \end{cases} \quad (3.9)$$

Par la définition de l'argmax, une solution du problème 3.9 est nécessairement une position



$(\Pi^{(1)}, \dots, \Pi^{(n)})$  vérifiant :

$$\forall i, \Pi^{(i)} = \arg \max_p a_g(p, \Pi^{(0)}, \dots, \Pi^{(n)})$$

, c'est-à-dire un point fixe de la fonction  $\mathbf{f}$  définie lors de la relaxation. Cependant, cette fonction peut admettre plusieurs points fixes et la relaxation peut donc converger vers un maximum local qui n'est pas forcément la solution optimale. On peut noter que si  $\mathbf{f}$  admet un unique point fixe et que la relaxation converge, alors elle converge vers l'unique solution du problème d'optimisation 3.9, qui maximise l'activité totale de l'architecture.

Maintenant que nous avons posé un formalisme sur l'algorithme, le but de ce chapitre est de visualiser la relaxation sur un exemple d'expérience. Nous observerons expérimentalement dans la suite du chapitre que lorsque les poids des cartes présentent une organisation au sens des cartes de Kohonen, la fonction  $\mathbf{f}$  apparaît admettre un unique point fixe, ce qui assurerait que la relaxation, si elle converge, converge bien vers la solution du problème 3.9. Par ailleurs, nous observerons que lorsque ce point fixe existe, la relaxation converge quelle que soit l'état initial des BMUs.

### 3.2.3 Relaxation et notion de *Best Matching Unit*

La notion de Best Matching Unit est définie au sein de l'algorithme d'apprentissage d'une carte de Kohonen comme l'unité possédant l'activité maximale pour une entrée fixée. Cette activité dépend de l'entrée et des poids de la carte. On attend d'un algorithme de recherche du BMU que la valeur trouvée en sortie soit uniquement relative aux poids de la carte et aux entrées : elle ne doit pas dépendre de l'initialisation. Dans le cas de la recherche de BMU par relaxation, nous voulons d'abord vérifier que la recherche du BMU converge. Dans notre modèle, nous arrêtons la relaxation soit lorsqu'elle a convergé, soit après un nombre fixé  $\tau_{max}$  d'itérations. Il est donc toujours possible de définir un BMU, mais dans le cas où la relaxation n'a pas convergé, ce BMU n'a pas le sens de « Best Matching Unit ». Pour vérifier cela, nous mesurerons la convergence de la relaxation sur un ensemble d'expériences. Nous observerons dans un second temps si la valeur trouvée à l'issue de la relaxation, à savoir le point de convergence s'il existe, ne dépend pas des conditions initiales de la relaxation. Le BMU sera alors relatif à seulement l'entrée et l'état des poids de la carte et non aux conditions initiales. Pour cela, nous chercherons à vérifier que la fonction  $\mathbf{f}$  générant la suite  $\Pi_\tau$  admet un unique point fixe sur un ensemble d'exemples.

Ce chapitre présente des points d'analyse empirique de ce processus de relaxation, réalisée sur des cartes 1D et 2D prenant en entrée un jeu de données 1D. L'évaluation de ces deux conditions sera réalisée sur des exemples d'apprentissage d'une architecture de deux et trois cartes. Notons que dans l'algorithme CxSOM, les BMUs sont initialisés à la position du maximum de l'acti-

→ Mélange de causes & conséquences la, non? Que regardent-ils qui permet de dire quoi?

Plutôt que de distinguer BMU et Best Matching Unit, BMU par rapport à l'opération de relaxation, quoi? sensibilité à l'initialisation, observer

du → vérifier

→ ?

chaque?

### 3.3. Étude expérimentale de la convergence de la relaxation

tivité externe dans chaque carte. Nous effectuerons l'analyse de la relaxation pour des conditions initiales quelconques, mais il est important de noter que ce choix d'initialisation permettra en fait d'éviter certains cas de non-convergence. Nous montrerons expérimentalement qu'il n'existe pas toujours un unique point fixe, en particulier lorsque les poids sont répartis aléatoirement au début de l'apprentissage, ce qui entraîne une non-convergence de la relaxation. Cependant, nous observerons que ces cas de non-convergence n'influencent pas le dépliement ultérieur des cartes, (ce qui est permis par l'initialisation de la relaxation à la position maximisant l'activité externe.) → ? Nous observerons expérimentalement qu'à la fin de l'apprentissage, la disposition des poids obtenue dans chaque carte permet l'existence d'un unique point fixe et que la relaxation converge vers ce point, indépendamment de l'initialisation des valeurs de  $\Pi_r^{(i)}$ . Enfin, nous observerons comment le choix du pas de relaxation  $\Delta$  influence la convergence de la relaxation.

### 3.3 Étude expérimentale de la convergence de la relaxation

figure 2.4?

➤ Nous nous intéressons ici à un processus d'apprentissage complet d'une architecture de deux cartes. Les entrées  $X^{(1)}$  et  $X^{(2)}$  que nous présentons à chaque carte sont les coordonnées  $x$  et  $y$  de points situés sur un cercle de centre 0.5 et de rayon 0.5. La disposition des entrées n'a pas d'importance ici, et nous détaillerons le choix de disposition d'entrées dans les chapitres suivants ; notons simplement que les entrées de chaque carte sont normalisées, et chaque entrée s'étend sur toutes les valeurs entre 0 et 1.

Pendant l'apprentissage, nous effectuons des phases de test régulières à des temps  $t$ , à poids figés, sur 5000 points tirés selon les mêmes distributions d'entrées. Nous comptons ensuite, pour chaque entrée de test réalisé au temps  $t$ , le nombre de pas nécessaires avant la convergence de la relaxation. Lors des tests, nous initialisons la relaxation à la position maximisant l'activité externe.

utile?

[En pratique, l'algorithme de relaxation s'arrête si la relaxation dépasse  $\tau_{max}$  ici fixé à 1000 itérations ;] nous considérerons que la relaxation n'a pas atteint un point de convergence si le nombre de pas de relaxation  $\tau$  atteint  $\tau_{max}$  itérations lors de l'expérience. À partir de ces valeurs, nous pouvons tracer le nombre de pas moyen nécessaires à la convergence (en prenant en compte les cas dans lesquels la relaxation ne converge pas), ainsi que le taux de convergence : la proportion d'entrées sur les 5000 entrées tests pour lesquelles la relaxation a convergé. Nous répétons 10 fois l'apprentissage complet et les tests, sur les mêmes distributions d'entrée. Nous réalisons ces expériences sur des cartes 1D et des cartes 2D. Nous traçons en figure 3.1 la moyenne et l'écart type du nombre moyen de pas de relaxation et du taux de convergence obtenues sur ces 10 répétitions. La figure 3.2 présente ces valeurs obtenues avec des cartes 2D.

Plusieurs situations peuvent se traduire par une non-convergence de la relaxation :

— La relaxation évolue vers un point de convergence, mais trop lentement pour y arriver en

moins de 1000 itérations

- La relaxation évolue vers un cycle limite composé d'un nombre réduit d'unités se succédant alternativement
- La relaxation évolue sans répétition d'un motif dans chaque carte ; il s'agit d'une évolution chaotique.

Le premier cas est évité car la limite de 1000 itérations est assez grande par rapport à la taille de la carte, les cartes sont de taille 500 et le pas d'évolution de la relaxation d'une dizaine d'unités. La convergence, si elle existe, est rapide. Les cas de non-convergence concernent alors la deuxième et la troisième situation. Nous observerons plus précisément les trajectoires dans la section suivante ; on s'intéresse ici seulement à la question de la convergence.

Les figures 3.1 et 3.2 montrent qu'au début de l'apprentissage, lorsque les poids sont initialisés aléatoirement, la relaxation atteint seulement un point de convergence dans 20% des cas. Lorsque les cartes sont dépliées, la relaxation évolue vers un point de convergence dans plus de 95% des cas pour des cartes 1D, et 90% pour des cartes 2D. L'évolution de la convergence est similaire pour des architectures de deux et trois cartes.

Nous en concluons que la convergence de la relaxation dépend complètement de la disposition des poids. Elle converge dans plus de 90 % des cas lorsque les poids sont bien dépliés, mais seulement rarement au début de l'apprentissage lorsque les poids ne présentent aucune forme de continuité et d'organisation. Toutefois, le fait que la relaxation ne converge pas en début d'apprentissage ne perturbe pas l'organisation des poids : on observe que la carte se déplie correctement au long de l'apprentissage. Ce comportement peut être expliqué. On observe que la relaxation converge bien à partir du moment où les poids externes sont organisés et présentent une continuité. Au début de l'apprentissage, même si la relaxation mène à des positions quelconques de BMUs, ces BMUs auront quand même des poids externes restant proches de la valeur de l'entrée externe. Le calcul de l'activité dépend en effet principalement de l'activité externe de la carte :

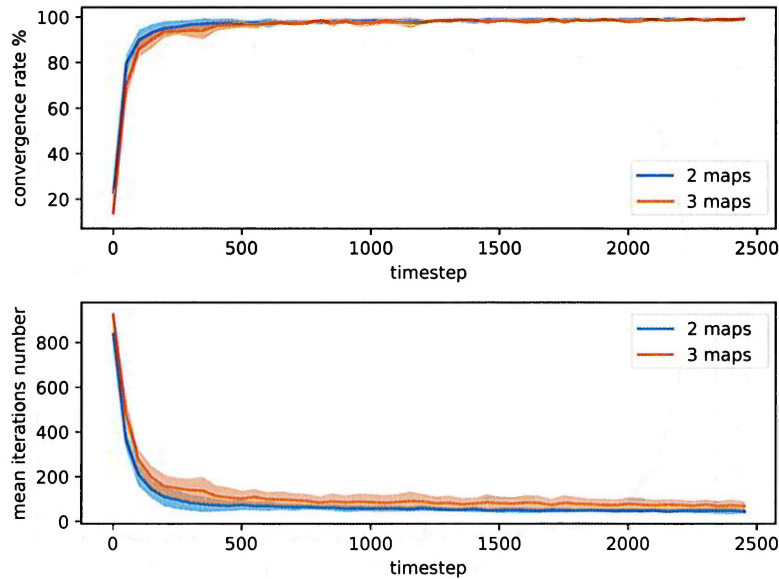
$$a_g = \sqrt{a_e \left( \frac{a_e + a_c}{2} \right)}$$

De plus, la relaxation est initialisée à une position correspondant au maximum de l'activité externe. Bien que le BMU n'aie pas de sens topologique dans la plupart des cas au début de l'apprentissage, les poids externes vont se déplier correctement sur les entrées externes. Le grand rayon de voisinage externe permet aux poids externes de se déplier plus rapidement que les poids contextuels. Une fois les poids externes dépliés, la relaxation apparaît converger sur les exemples étudiés. Le dépliement des poids contextuels, plus lent, s'effectue donc sur des BMUs ayant un « sens ».



un mdr pour  
expliquer 2 & 3 cartes?

### 3.4. Étude de l'évolution de la relaxation



en anglais?

point de départ  
citato

FIGURE 3.1 – En haut : évolution de la moyenne et l'écart-type du taux de convergence de la relaxation au cours de l'apprentissage sur deux et trois cartes 1D. En bas : évolution du nombre moyen de pas nécessaires à la convergence de la relaxation. Chaque point est calculé sur un échantillon de 5000 relaxations au temps  $t$ , évaluées sur des entrées différentes prises aléatoirement sur le cercle. La moyenne et l'écart-type sont réalisés sur 10 apprentissages séparés.

## 3.4 Étude de l'évolution de la relaxation

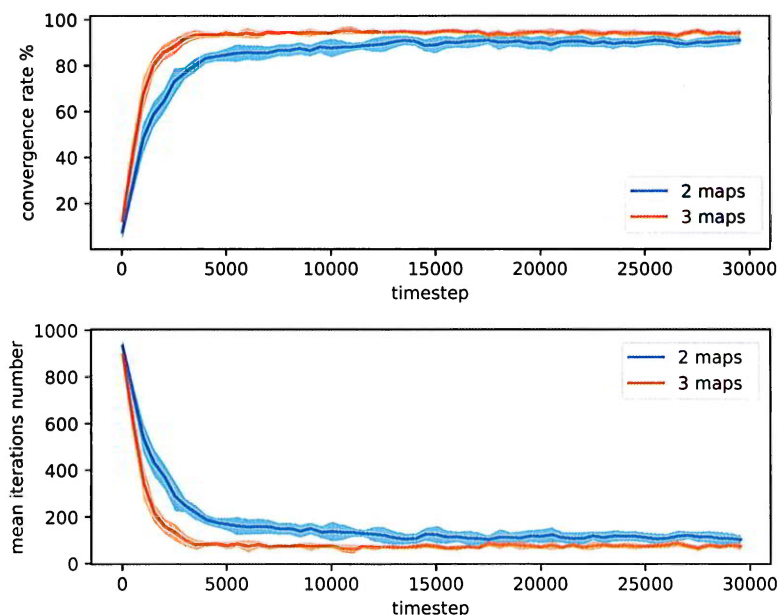
### 3.4.1 Trajectoires de relaxation

Nous nous plaçons dans une architecture de deux cartes 1D. Nous étudions dans cette partie l'évolution de plusieurs processus de relaxation lancés sur des poids de cartes dans une même disposition, [à présent] en fixant l'entrée externe et en prenant des valeurs d'initialisation de relaxation  $\Pi_0$  différentes. Nous voulons vérifier comment la relaxation évolue en fonction de l'initialisation des BMUs. Nous avons vu qu'au début de l'apprentissage, la relaxation ne converge pas : nous tracerons l'évolution des trajectoires de relaxation dans le cas où les poids des cartes sont aléatoires ( $t = 0$ ). Nous effectuerons les mêmes tracés dans le cas où les poids externes et contextuels des cartes sont dépliés. Nous avons vu que la relaxation converge dans la majorité des cas ; nous regarderons s'il existe un unique point de convergence en fonction des conditions initiales de la relaxation.

initialisés?

Pour une architecture de deux cartes 1D, nous avons :

$$\begin{cases} \hat{p}_\tau^{(1)} = \arg \max_p (a_g^{(1)}(p, \Pi_\tau^{(2)})) \\ \hat{p}_\tau^{(2)} = \arg \max_p (a_g^{(1)}(p, \Pi_\tau^{(1)})) \end{cases}$$



laine  
&  
anglais?

FIGURE 3.2 – En haut : évolution de la moyenne et l'écart-type du taux de convergence de la relaxation au cours de l'apprentissage sur deux et trois cartes 2D. En bas : évolution du nombre moyen de pas nécessaires à la convergence de la relaxation. Chaque point est calculé sur un échantillon de 1000 relaxations au temps  $t$ , évaluées sur des entrées différentes prises aléatoirement sur le cercle. La moyenne et l'écart-type sont réalisés sur 10 apprentissages séparés.

et

est possible

$$\begin{cases} \Pi_{\tau+1}^{(1)} = \Pi_{\tau}^{(1)} \pm \text{sgn}(\hat{p}_{\tau}^{(1)} - \Pi_{\tau}^{(1)}) \\ \Pi_{\tau+1}^{(2)} = \Pi_{\tau}^{(2)} \pm \text{sgn}(\hat{p}_{\tau}^{(2)} - \Pi_{\tau}^{(2)}) \end{cases} \quad ? \quad \pi_{\tau}^{(1)} \pm \Delta ?$$

Avec  $\hat{p}^{(1)}$  dépendant seulement de  $\Pi^{(2)}$  et inversement. Dans cette situation, nous pouvons tracer la trajectoire de  $\Pi_{\tau}$  dans l'espace  $(p^{(1)}, p^{(2)})$ .

avec?

⌘ Nous nous intéressons, dans chaque configuration de poids, aux trajectoires et aux points d'arrivées de 200 relaxations, obtenus pour une même entrée  $X$ , mais des valeurs d'initialisation  $\Pi_0$  différentes choisies aléatoirement parmi [toutes] les positions des cartes. Pour un problème à deux cartes, il est peu coûteux de trouver, par une recherche exhaustive, les points fixes de la fonction  $f$  générant la suite des BMUs. Nous visualiserons ainsi les trajectoires de relaxation dans l'espace  $(p^{(1)}, p^{(2)})$  superposées à la valeur, selon  $(p^{(1)}, p^{(2)})$ , de  $|\hat{p}^{(1)} - p^{(1)}|$  et  $|\hat{p}^{(2)} - p^{(2)}|$ . Nous regarderons s'il existe un unique point où ces deux quantités s'annulent, c'est à dire le point fixe de  $f$ , et vérifierons si les trajectoires de relaxation convergent vers ce point.

archi?

qui détermine?

En figure 3.3, nous traçons les deux quantités  $|\hat{p}^{(1)} - p^{(1)}|$  et  $|\hat{p}^{(2)} - p^{(2)}|$  au début de l'apprentissage. Les positions annulant chacune des quantités sont marquées par les zones en violet

différentes

→ Comme c'est possible, nous visualisons...

sur chaque graphique. Nous y faisons figurer uniquement les positions de fin de relaxation de  $(\Pi^{(1)}, \Pi^{(2)})$  pour plus de lisibilité. Ces positions sont marquées par les points noirs. Nous remarquons une position en  $(0.1, 0.75)$ , dans laquelle les deux quantités sont nulles. Ce point fixe apparaît comme un hasard du choix des poids. Nous observons que certaines trajectoires ont convergé vers cette position : nous observons des points noirs à cet emplacement, mais que d'autres trajectoires ont amené la suite des BMUs vers d'autres positions. (Nous observons la présence de plusieurs cycles limites : certaines trajectoires ne convergent pas et évoluent sur un cycle de positions.) Les points noirs obtenus sur ces cycles sont en fait les dernières positions atteintes au temps  $\tau_{max}$  de la relaxation. Afin de mieux représenter les trajectoires, nous traçons également en 3.5 le champ des déplacements à effectuer de  $(\Pi_\tau^{(1)}, \Pi_\tau^{(2)})$  lors de la relaxation en fonction de la position courante. Nous y superposons les trajectoires aboutissant aux points présentés en figure 3.3. Ces champs de déplacement permettent d'observer les comportements de point fixe et cycles limites des suites de BMUs.

La configuration en fin de l'apprentissage est présentée en figure 3.4. Nous y ajoutons cette fois 10 exemples de trajectoires de BMUs. Une structure apparaît dans les valeurs des différences, et un point où les deux différences sont nulles existe à l'intersection des deux zones violettes. Sur ces tracés, nous observons un seul point d'attraction pour les 200 trajectoires. La relaxation ne dépend donc plus des conditions initiales. Ce point correspond à la position où les deux différences  $|\hat{p}^{(1)} - p^{(1)}|$  et  $|\hat{p}^{(2)} - p^{(2)}|$  sont nulles. La figure 3.6 présente le champ de déplacement des BMUs en fonction de la position courante. Les trajectoires suivent les zones où l'une ou l'autre des différences est nulle, pour mener à la position stable. Nous avons observé cette configuration pour plusieurs entrées. À la fin de l'apprentissage, nous observons sur ces exemples que la fonction  $f$  admet un unique point fixe, donc que la recherche de maximum de l'activité totale de l'architecture admet une solution unique. Dans ce cas, la relaxation converge quelles que soient les conditions initiales. Les BMUs trouvés sont alors relatifs à la disposition du poids des cartes et des entrées et non à des conditions internes aux cartes, donnant au BMU  $(\Pi^{(1)}, \Pi^{(2)})$  un sens : il s'agit d'une position dépendant seulement de l'entrée  $(X^{(1)}, X^{(2)})$  et correspond au maximum de l'activation totale des cartes de l'architecture.

Nous pensons que l'existence du point fixe vient de la disposition ordonnée des poids externes à la fin de l'apprentissage et de la formule du calcul d'activité. À la fin de l'apprentissage, les poids externes sont organisés et  $a_e(X, p)$  présente un seul pic d'activation. De plus, l'activité globale correspond à la modulation de l'activité contextuelle par les valeurs de l'activité externe. Nous observons qu'à  $X^{(1)}$  fixé,  $\hat{p}^{(1)} = \arg \max_p a_g(p^{(1)}, p^{(2)})$  reste dans une zone restreinte lorsque  $p^{(2)}$  varie (de même pour  $a_g^{(2)}$ ). La relaxation évoluera forcément vers cette zone réduite de la carte. Cette propriété est illustrée en figure 3.7. Nous y faisons figurer les poids des deux cartes et les valeurs de  $\hat{p}^{(1)}$  et  $\hat{p}^{(2)}$  qu'on obtient à entrée externe fixée  $X$  et en faisant varier  $p^{(1)}, p^{(2)}$  dans tout l'espace des positions possibles. Quelle que soit l'entrée contextuelle,  $\hat{p}^{(1)}$  sera ainsi forcément situé entre 0.26 et 0.34 pour une même entrée  $X^{(1)}$  présentée et la configuration de



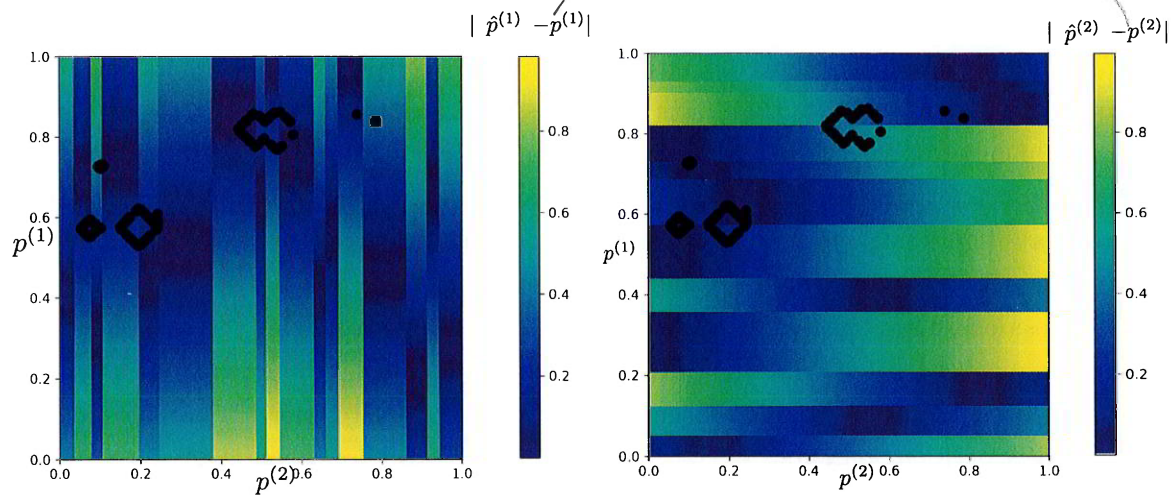


FIGURE 3.3 – Valeur de  $\hat{p}^{(1)} - p^{(1)}$ , resp.  $\hat{p}^{(2)} - p^{(2)}$  à  $t = 0$ , lorsque les poids sont encore aléatoirement disposés dans chaque carte.  $\hat{p}^{(1)}$  ne dépend que de  $p^{(2)}$  : on peut donc tracer cette valeur selon deux dimensions pour chaque carte. Les zones où cette valeur est nulle sont en violet sur le graphique. Les points fixes, s'il existent, sont aux positions de différence nulle pour  $M^{(1)}$  et  $M^{(2)}$ . Les points noirs représentent les points de convergence pour 200 trajectoires de relaxation, lancées pour différents  $\Pi_0^{(1)}, \Pi_0^{(2)}$ .

poids courante de la carte.

D'après cet exemple, que nous avons observé sur de nombreuses entrées, nous formulons l'hypothèse que le dépliement ordonné des poids externes dans une carte en 1D est une condition suffisante à l'existence d'un unique maximum d'activité. Dans ce cas, la relaxation permet de trouver ce point fixe.

### 3.4.2 Influence du pas de relaxation

Dans les expériences précédentes, nous avons utilisé un pas de convergence  $\Delta = 0.05$ . Une autre solution est de ne pas utiliser de pas de relaxation, c'est à dire, à chaque itération, déplacer le BMU  $\Pi_\tau^{(i)}$  directement en  $\hat{p}^{(i)}$ , où l'activité globale est maximale, au lieu de le déplacer de  $\Delta$ .

L'évolution de la relaxation devient alors :

$$\forall i, \Pi_{\tau+1}^{(i)} = \hat{p}_\tau^{(i)} \quad (3.10)$$

On pourrait supposer que prendre un petit  $\Delta$  permet une meilleure convergence ; en fait, la valeur de  $\Delta$  influence peu la capacité de convergence et l'organisation des cartes. En figure 3.8, nous avons tracé plusieurs trajectoires de relaxation réalisées après apprentissage, pour une même entrée. Nous représentons sur cette même figure les différences  $|\hat{p}^{(1)} - p^{(1)}|$  et  $|\hat{p}^{(2)} - p^{(2)}|$ . La méthode de relaxation utilisée lors de l'apprentissage menant à la configuration de poids utilisée



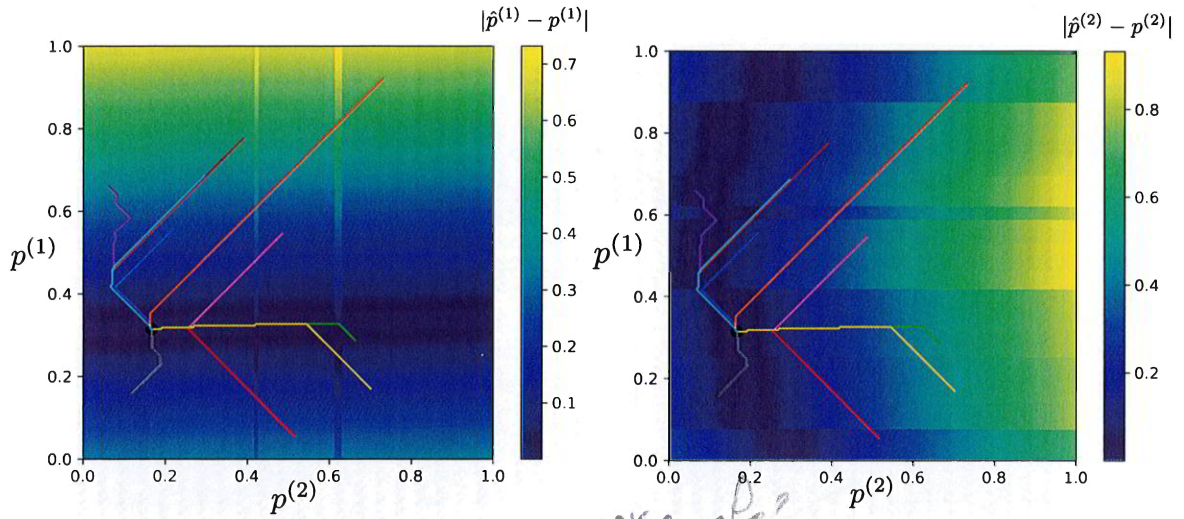


FIGURE 3.4 – Valeur de  $|\hat{p}^{(1)} - p^{(1)}|$ , resp.  $|\hat{p}^{(2)} - p^{(2)}|$ , lorsque les cartes sont organisées telles que représentées en figure 3.7.  $\hat{p}^{(1)}$  ne dépend que de  $p^{(2)}$  : on peut donc tracer les quantités ci-dessus selon les deux dimensions  $p^{(1)}, p^{(2)}$  pour chaque carte. Les zones où cette valeur est nulle sont en violet sur le graphique. Les points fixes, s'il existent, sont aux positions de différence nulle à la fois pour  $M^{(1)}$  et  $M^{(2)}$ . Les points noirs représentent les points d'arrivée de la relaxation pour 200 trajectoires de relaxation, lancées pour différents  $\Pi_0^{(1)}, \Pi_0^{(2)}$ , dont 10 sont tracées sur le graphique. Toutes ces trajectoires de relaxation convergent vers un même point qui est l'unique point fixe de la fonction de relaxation  $p^{(1)}, p^{(2)} \rightarrow |\hat{p}^{(1)} - p^{(1)}|, |\hat{p}^{(2)} - p^{(2)}|$

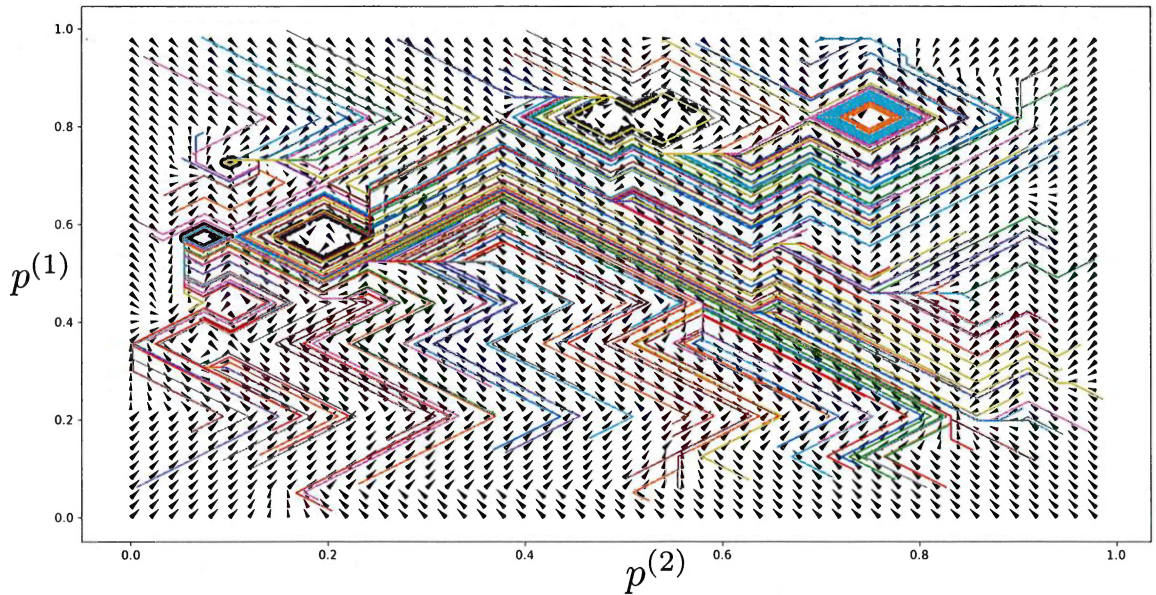


FIGURE 3.5 – Champ de déplacements de  $\Pi_r^{(1)}, \Pi_r^{(2)}$  lorsque les poids sont aléatoires, à  $t = 0$ . Nous représentons les trajectoires de 200 relaxations initialisées différemment. En fonction de la position initiale des BMUs, la relaxation évolue vers un point fixe ou un cycle limite.

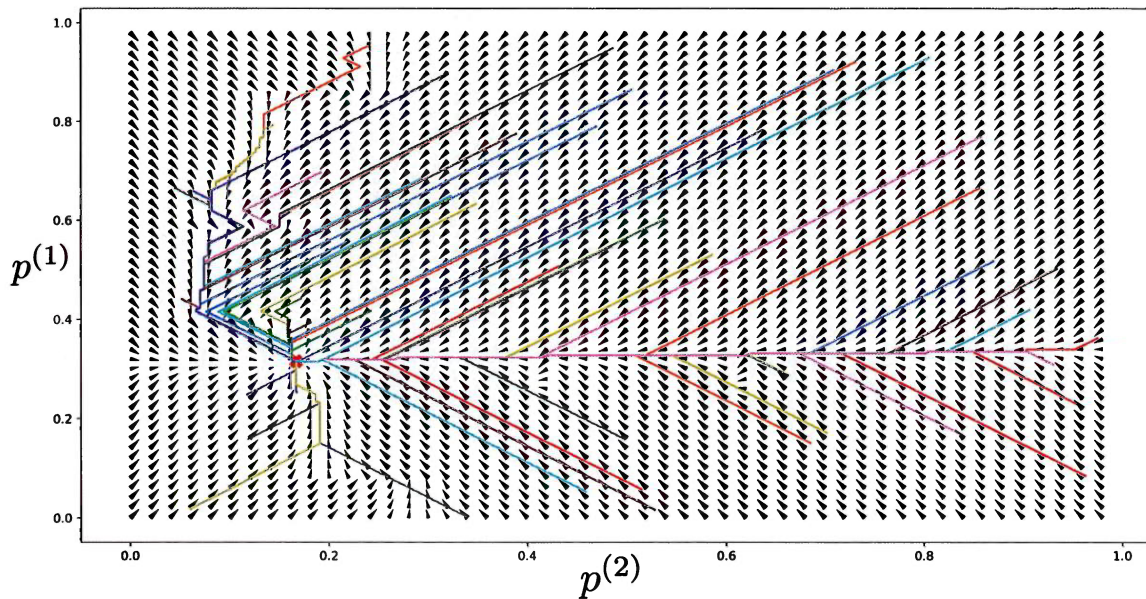


FIGURE 3.6 – Champ de déplacements de  $\Pi^{(1)}, \Pi^{(2)}$  lorsque les poids sont organisés tels que représentés en figure 3.7, à  $t = 9999$ . Nous y représentons les trajectoires suivies par 50 relaxations initialisées différemment. Les relaxations évoluent vers un point fixe commun. → coord?

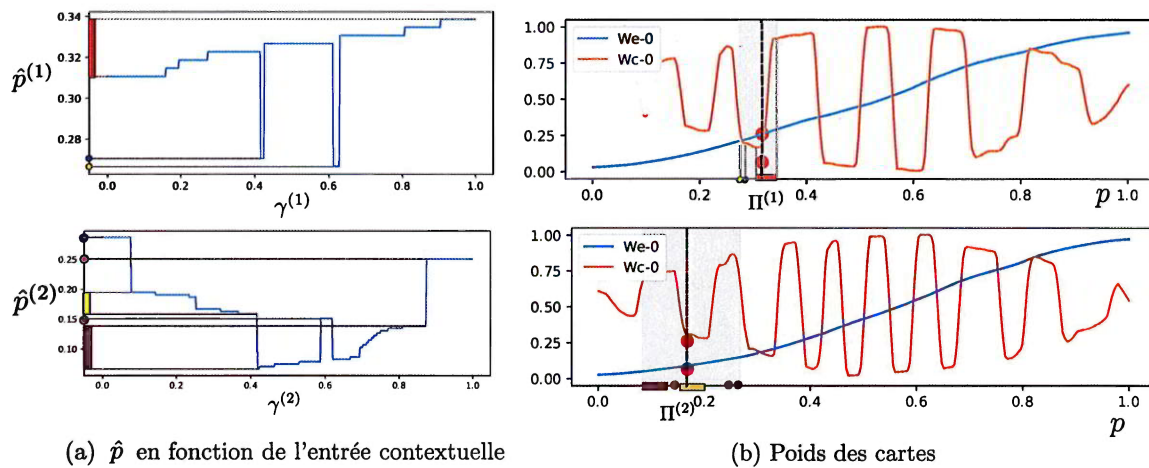


FIGURE 3.7 – (a) :  $\hat{p}^{(1)}$  et  $\hat{p}^{(2)}$  en fonction de l'entrée contextuelle de leur carte  $\Pi^{(2)}$  et  $\Pi^{(1)}$ . (b) : les poids externes et contextuels des cartes 1 et 2 sont représentés selon leur position dans la carte. On représente également les entrées test  $X^{(1)}$  et  $X^{(2)}$  en fonction de leur BMU. Les entrées utilisées pour tracer les figures de gauche sont colorées en rouge sur les figure de droite :  $X^{(1)} = 0.26, X^{(2)} = 0.06$ . Les intervalles dans lequel les valeurs de  $\hat{p}$  varient sont reportés sur la figure (b). passé à la figure.



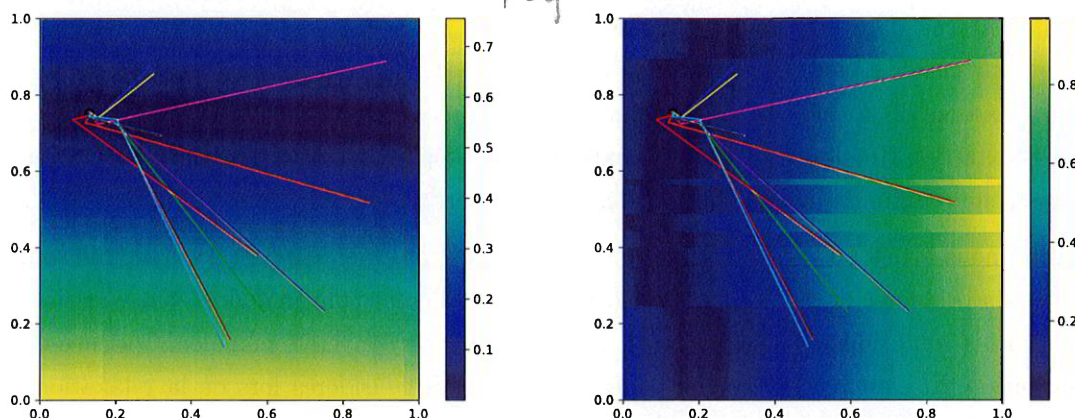


FIGURE 3.8 – Trajectoires des relaxations ( $\Pi_\tau^{(1)}, \Pi_\tau^{(2)}$ ) dans le champ des différences  $\hat{p}^{(1)} - p^{(1)}$  et  $\hat{p}^{(2)} - p^{(2)}$ , lorsque la relaxation est effectuée sans utiliser de petits déplacements. Les tracés sont effectués après apprentissage. La relaxation semble encore converger vers un point fixe.

pour cette expérience n'a pas non plus utilisé de pas de relaxation  $\Delta$  à l'issue de l'apprentissage, le comportement de la relaxation ne change pas par rapport à celui observé en figure 3.4 : les trajectoires convergent vers le point fixe de  $f$ . Cela se comprend au vu des résultats précédents.  $\hat{p}^{(1)}$  et  $\hat{p}^{(2)}$  sont dans un intervalle réduit de valeurs de la carte, quelles que soient les positions  $p^{(1)}$  et  $p^{(2)}$ . Amener directement le BMU dans cet intervalle ou l'y amener pas à pas change peu le comportement de la suite.

conclure donc ?

### 3.5 Conclusion

Ce chapitre présente une étude expérimentale du mécanisme de relaxation. Nous avons détaillé un formalisme décrivant l'algorithme, qui montre que la relaxation est une recherche du maximum de l'activation globale de l'architecture, soit la somme des activations de chaque carte. La relaxation est alors un moyen de trouver un ensemble de BMU au sein d'une architecture maximisant une propriété *globale* à cette architecture : toutes les cartes voient leur activité globale maximisée. Cette recherche de maximum est réalisée localement, au niveau de chaque carte, et non de façon globale. La relaxation agit alors comme une manière de connecter des cartes de façon non-hiérarchique et les BMUs sont l'interface entre modules de la carte.

La convergence de la relaxation dépend de la disposition des poids des cartes. Nous avons observé expérimentalement que lorsque les poids des cartes ne sont pas ordonnés, il n'existe pas forcément de solution au problème d'optimisation. Dans ce cas, la relaxation converge seulement dans 20% des cas lors d'une étape de test. Cependant, nous avons observé que les cartes se déplient ~~correctement~~ même lorsque la relaxation n'a pas convergé, ce qui est assuré grâce à l'initialisation de la relaxation à des positions maximisant l'activité externe et grâce à la prépondérance de

l'activité externe dans le calcul de l'activité globale. La relaxation va alors évoluer vers un point qui, même s'il n'est pas une solution du problème de maximisation globale, maximise bien l'activité externe de la carte :  $\omega_e(\Pi)$  est proche de l'entrée. Cela permet de déplier correctement les poids externes. Une fois que ces poids externes sont dépliés, la relaxation ne peut évoluer que vers une zone réduite de la carte qui est définie par l'activité externe.

Enfin, nous avons observé que lorsque tous les poids sont bien dépliés, il existe une solution optimale du problème de relaxation. L'algorithme de recherche converge alors vers ce point quel que soient les conditions initiales. Dans ce cas, la relaxation converge même en l'absence de pas de relaxation  $\Delta$ , c'est à dire lorsque les BMUs successifs considérés lors de la relaxation sont directement les positions maximisant l'activité de chaque carte.

Ces tracés de relaxation ont été effectués pour des cartes 1D prenant des entrées 1D, avec des cartes ayant un jeu de paramètres spécifique, avec notamment  $r_e = 0.2$  et  $r_c = 0.02$ . Nous avons également observé que la relaxation converge sur des cartes 2D prenant des entrées 1D. Dans les chapitre suivants, nous détaillerons le comportement des cartes 1D et 2D sur différentes données d'entrées et différents paramètres. Nous prendrons soin de tracer dans ces chapitres l'évolution de la convergence de la relaxation, d'après la méthode proposée en figure 3.1 ; nous observerons que la relaxation semble bien converger dès que les poids présentent un dépliement ordonné, ceci sur des cartes 1D prenant des entrées 1D mais également des cartes 2D prenant des entrées 2D au chapitre 7.