



A propos des architectures de cartes auto-organisatrices stylées

THÈSE

présentée et soutenue publiquement le le plus tard possible en 2022

pour l'obtention du

Doctorat CentraleSupélec

(mention informatique)

par

Noémie Gonnier

Composition du jury

<i>Président :</i>	Le président	du jury
<i>Rapporteurs :</i>	Le rapporteur 1	du laboratoire
	Le rapporteur 2	
	Le rapporteur 3	
<i>Examineurs :</i>	L'examineur 1	
	L'examineur 2	

Laboratoire Lorrain de Recherche en Informatique et ses Applications

Mis en page avec la classe thesul.

Sommaire

1	Approche modulaire des réseaux de neurones	1
1.1	Contexte	1
1.1.1	Systèmes complexes	2
1.2	Quelle définition de la modularité?	3
1.2.1	Modularité structurelle	3
1.2.2	Modularité fonctionnelle	5
1.2.3	Modularité temporelle	6
1.2.4	Modularité émergente ou modularité définie	6
1.2.5	Emergence d'une modularité et auto-organisation	6
1.3	La modularité, répandue dans les système biologiques	6
1.3.1	Le cerveau, réseau modulaire fondamental	6
1.3.2	Des réseaux et des modules dans tous les systèmes biologiques	7
1.4	Intérêt computationnel des réseaux modulaires	8
1.4.1	Modularité, complexité et émergence d'un apprentissage?	8
1.4.2	Types de connexions	8
1.5	Modularité dans les réseaux de neurones	8
1.5.1	Deep Learning	9
1.5.2	Réseaux auto-organisés	9
1.5.3	Construire une architecture modulaire	9
1.6	Enjeux d'une architecture modulaire de SOMs	10
2	Un réseau de neurones auto-organisé : les cartes de kohonen	13
2.1	Principe Général	13
2.2	Approche topologique des cartes de Kohonen	13
2.3	Principes d'organisation	13
2.4	Travaux préliminaires sur les architectures de cartes de Kohonen	13
2.5	A trier	13

3	Construction d'une architecture modulaire de cartes auto-organisatrices	15
3.1	Description de l'algorithme	15
3.1.1	Carte de Kohonen classique	15
3.1.2	Modèle : CxSOM	16
3.2	Choix des paramètres	19
3.2.1	Influence des rayons de voisinage	19
3.2.2	Influence des autres paramètres	19
3.2.3	Compatibilité en 2D	19
3.3	Analyse de la relaxation	19
3.3.1	Analyse expérimentale	19
3.3.2	Champs de BMU	19
3.3.3	Limitations et possibilités en grande dimension	19
3.4	Implémentation	19
3.5	Perspectives d'évolutions	19
4	Analyser l'organisation : une approche par variables aléatoires	21
4.1	Cas d'utilisation : les entrées multimodales	21
4.1.1	Définition et inspiration biologique	21
4.1.2	Formalisme	21
4.1.3	Perspectives	21
4.2	Représentation des entrées	21
4.3	Information apprise par une carte	21
4.4	Représenter une carte au sein d'une architecture	22
4.5	Prédiction d'entrée	24
	Bibliographie	25

Introduction

Cette thèse propose une contruction d'une architecture modulaire

Chapitre 1

Approche modulaire des réseaux de neurones

Sommaire

1.1	Contexte	1
1.1.1	Systèmes complexes	2
1.2	Quelle définition de la modularité ?	3
1.2.1	Modularité structurelle	3
1.2.2	Modularité fonctionnelle	5
1.2.3	Modularité temporelle	6
1.2.4	Modularité émergente ou modularité définie	6
1.2.5	Emergence d'une modularité et auto-organisation	6
1.3	La modularité, répandue dans les système biologiques	6
1.3.1	Le cerveau, réseau modulaire fondamental	6
1.3.2	Des réseaux et des modules dans tous les systèmes biologiques	7
1.4	Intérêt computationnel des réseaux modulaires	8
1.4.1	Modularité, complexité et émergence d'un apprentissage ?	8
1.4.2	Types de connexions	8
1.5	Modularité dans les réseaux de neurones	8
1.5.1	Deep Learning	9
1.5.2	Réseaux auto-organisés	9
1.5.3	Construire une architecture modulaire	9
1.6	Enjeux d'une architecture modulaire de SOMs	10

1.1 Contexte

L'idée de ce chapitre est d'inscrire nos travaux dans le coté modulaire des réseaux de neurones. Il nous faudra donc définir proprement ce qu'on appelle modularité, et définir les motivations pour créer une architecture modulaire. Il faut def ce qu'on appelle réseau de neurones et apprentissage automatique.

D'une part, de nombreux modèles biologiques présentent des architectures modulaires. Notre conception du monde se présente en fait sous une forme de modularité. En tant qu'humain, nous comprenons le monde de notre point de vue, en le décomposant pour qu'il semble accessible : en effet notre raisonnement est modulaire, notre système social, groupes d'individus, etc, comme

l'explique par exemple [14] dans "How a mind resides in the brain". Nos constructions sont modulaires : programme informatique ... Il est difficile de concevoir et surtout de comprendre, en tant qu'humain, un système qui ne serait pas décomposable en modules. Prenons comme exemple les réseaux de neurones profonds : la compréhension et l'interprétabilité ces programmes est un défi de la recherche actuelle. Pour cette interprétation, on cherche des éléments symboliques, des groupes, des communautés. La décomposition des sciences elles même, par exemple, nous permet de trouver des solutions aux problèmes a des échelles différentes. Le programmeur n'a pas besoin de comprendre en profondeur quels transistors composent les circuits ; l'expert.e en électronique n'a pas besoin de d'abord résoudre les équations qui régissent les mouvements ioniques au sein des transistors pour concevoir des circuits, etc. Seuls les principes régissant les comportement globaux d'un système comme le transistors ont besoins d'être connus pour utiliser ces systèmes dans une tâche ; tâche qui constituera également un module dans son ensemble et qui sera régie par des principes généraux, du transistor à l'utilisation d'un logiciel de dessin. Mais est ce que cette hiérarchie modulaire est essentiellement subjective ? A priori non. Une organisation modulaire est présente et calculable dans de nombreux systèmes.

Présentation d'exemples de théories dérivées d'une modularité bio-inspirée. Brooks [4]

Les modules sont également associés aux systèmes complexes. De nombreux travaux sont ainsi réalisés à la frontière entre domaines, Réseaux associés aux systèmes complexes, interactions.

1.1.1 Systèmes complexes

Argument : les systèmes complexes montrent des propriétés d'émergence de comportement. Statiquement, cela peut être la construction de fractales, ou dynamiquement des systèmes régis par des équations non linéaires. Ex synchronisation dans les réseaux : lucioles, oiseaux, et surtout cerveau. L'émergence de propriétés est difficile à prédire et exprimer par la nature des systèmes. un système complexe s'étudie en effet en le simulant. On a l'état initial et les équations qui régissent la dynamique (passage d'un état à un autre, ou en version continue), mais il est trop compliqué d'exprimer sous forme d'équation un état courant. Il est donc judicieux d'explorer et de simuler des nouveaux systèmes complexes, car on ne peut pas vraiment prédire d'ores et déjà un comportement.

→ Auto-organisation definition

Les phénomènes d'auto-organisation sont déjà un élément caractéristique des systèmes complexes : Alan Turing montre qu'à partir d'un état presque uniforme de composants chimique, leur réaction et diffusion crée des motifs répétés. Ces motifs sont créés à partir de règles locales, on est bien sûr un phénomène auto-organisé. Plus surprenant ces motifs appelés "turing patterns" sont présents dans de nombreux systèmes biologiques, par exemple sur la peau des animaux. Cela montre la prévalence des phénomènes d'auto-organisation dans la nature.

À partir de règles locales, des comportements globaux peuvent donc émerger, de façon auto-organisée. Un des exemples biologiques montrant cette complexité est ainsi le cerveau humain.

Qui dit système complexe dit système modulaire ? Réseaux sociaux, communautés de fourmis, autant d'exemples montrant une émergence auto-organisée de modules.

Système complexe adaptatif (cas) introduit par John Holland et Murray Gellmann. Ces systèmes sont un sous ensemble des systèmes complexes. Il s'agit de systèmes multi-agents qui réagissent à leur environnement, et s'auto-influencent. Ils se distinguent par l'émergence de propriétés et l'auto-organisation. C'est précisément ce qu'on cherche à faire dans un réseau de neurones. On dira qu'il y a eu un apprentissage si l'évolution du système a conduit à l'émergence de propriétés montrant une généralisation d'information sur les données présentées.

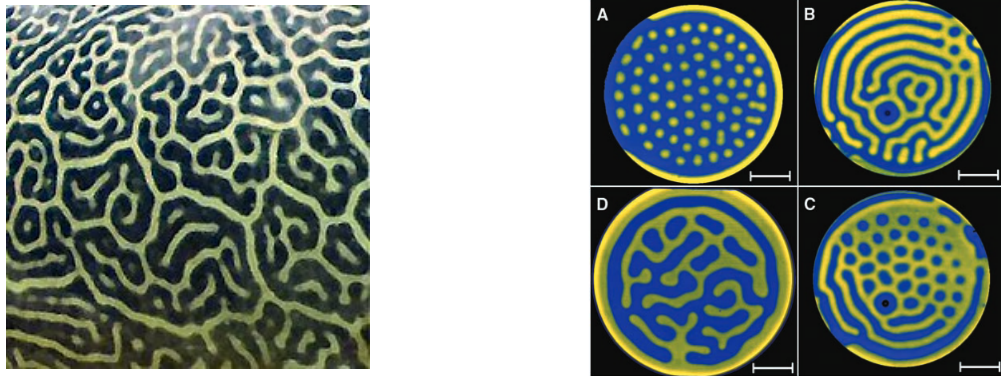


FIGURE 1.1 – Motifs de turing sur la peau du poisson-globe, à gauche. A droite, motifs de turing émergeant d'une réaction chimique [8].

1.2 Quelle définition de la modularité ?

L'étude de la modularité des systèmes est vaste. Entre étude des systèmes biologiques, ingénierie des réseaux de neurones, ou même sociologie, les définitions de modularité varient. Nous chercherons donc dans cette partie à exhiber des définitions et des spécificités de ce qu'on appelle modularité.

1.2.1 Modularité structurelle

Lorsque le système possède une structure définie de réseau, typiquement un réseau de neurones, on peut définir une modularité en terme de graphe. Un système modulaire est ainsi *Un système qui a une structure de graphe modulaire*. Même en tant que graphe, modulaire est un terme large. De nombreux travaux proposent "une architecture modulaire" sans vraiment définir ce terme.

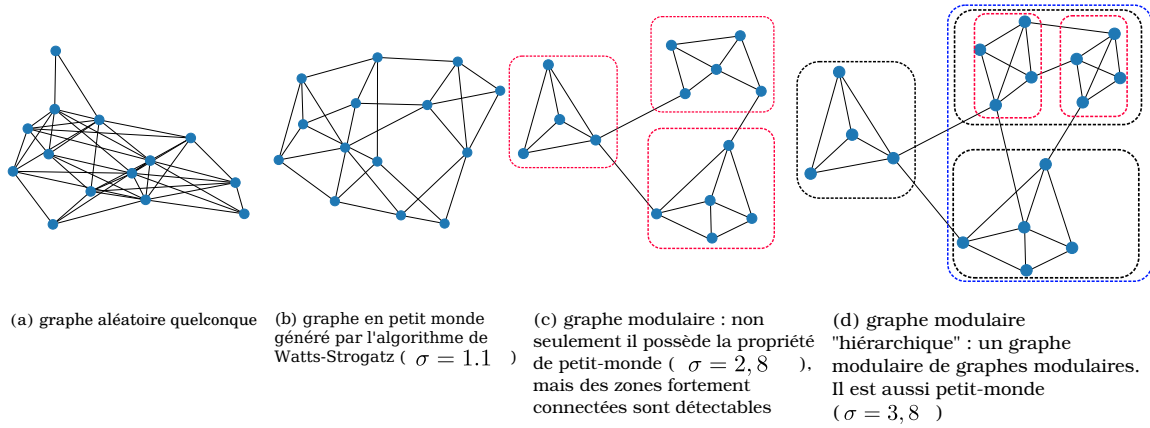
Mesurer la modularité structurelle d'un réseau

La modularité d'un réseau est définie en théorie des graphes par le fait de pouvoir détecter des zones fortement connectées au sein de ce réseau, reliées par peu de connexions. Il s'agit donc de détecter des cliques, des sous-graphes fortement connectés, et de les différencier des zones moins connectées. La quantité la plus largement utilisée pour définir la modularité d'un graphe est le *coefficient de clustering*. Ce coefficient mesure la probabilité que deux noeuds soient directement connectés en sachant qu'ils ont un voisin en commun. Les réseaux sociaux par exemple, présentent des forts coefficients de clustering. D'autres mesures sont possibles, menant à une détection de zones fortement connectées. Cette modularité peut également se mesurer en regardant la densité des arêtes dans une partition du graphe, relativement à ce qu'on attendrait dans un graphe aléatoire.

rich club -> ???

Réseaux en "petit-monde"

La modularité d'un réseau est liée à la propriété de petit-monde d'un graphe. Un graphe en *petit-monde* (small-world network) est un graphe dans lequel la distance moyenne entre deux noeuds est proportionnelle à $\log(N)$, N étant le nombre de noeuds du graphe. En d'autres



termes, c'est un graphe dans lequel on trouvera forcément un chemin assez court relativement à la taille du réseau, entre n'importe quels noeuds. Un exemple typique de réseau en petit monde est celui des relations sociales avec la règle des "six degrés de séparation" mis en lumière par Stanley Milgram en 1967 [13] : en prenant deux individus au hasard aux états unis, Milgram montre qu'on peut les relier de connaissance mutuelle en connaissance mutuelle en, en moyenne, 6 étapes. Maintenant que la plupart de nos connaissances sont en enregistrées sur les réseaux sociaux, cette distance, entre tous les utilisateurs de facebook dans le monde entier, a été mesurée comme étant 3.5 degrés de séparation en 2016¹. La propriété de petit-monde est mesurée par le *coefficient de petit-monde* σ . Il se mesure en comparant des métriques du graphe avec celles d'un graphe aléatoire équivalent, c'est à dire un graphe aléatoire ayant le même nombre de noeuds et la même densité de connexions. En notant C le coefficient de clustering du graphe, C_r celui d'un graphe aléatoire équivalent, L la longueur (nombre d'arête) moyenne du plus court chemin entre tous les noeuds du graphe, et L_r cette longueur dans un graphe aléatoire équivalent.

$$\sigma = \frac{\frac{C}{C_r}}{\frac{L}{L_r}}$$

Un réseau est dit petit-monde si $\sigma > 1$, autrement dit, si la longueur du plus court chemin est inférieure à celle d'un graphe aléatoire, et/ou si le graphe présente plus de communautés qu'un graphe aléatoire. Cette métrique est cependant assez limitée, et il est plus judicieux d'exister d'autres mesures, comparant cette fois le coefficient de clustering et la longueur moyenne du plus court chemin à des graphes équivalents en treillis en plus des graphes aléatoires.

Un réseau petit-monde n'est pas forcément modulaire. Par contre, un graphe ayant une structure modulaire, avec des communautés fortement connectées, est petit-monde. Un exemple est donné en figure 1.2.1.0. Le réseau (b) est petit-monde : on trouve forcément un chemin court entre deux noeuds. Le réseau (c) est également petit-monde, mais présente aussi des sous-graphes fortement connectés. La propriété de petit-monde semble avoir des avantages computationnel et se retrouve ainsi dans de nombreux exemples de réseaux biologiques. Nous détaillerons cet avantage en partie suivante.

1. <https://research.fb.com/three-and-a-half-degrees-of-separation>

Modularité auto-similaire

Si on prend un sous-graphe fortement connecté d'un graphe modulaire, ce sous-graphe peut éventuellement, à son tour, présenter une structure modulaire. On peut alors parler de modularité *hiérarchique*, ou *auto-similaire*. L'auto-similarité renvoie au processus de construction dans lequel on connecte par une arête deux structures jumelles (par exemple deux graphes complets) ; on copie ce graphe et on le connecte par une arête à sa copie pour former un graphe plus grand, etc. On gardera le terme auto-similaire pour parler de graphes formés de modules de modules, même si ces modules ne sont pas des copies exactes d'eux mêmes à différentes échelles, afin de garder le terme hiérarchique pour d'autres structures. Cette modularité "à différentes échelles" est souvent présente dans les réseaux. Le cerveau est par exemple souvent présenté comme un réseau modulaire auto-similaire [12]. Des modules larges (aires fonctionnelles) sont formés de sous-modules, eux mêmes décomposables en sous-modules, etc.

Réseaux invariant par échelle

Un dernier type de réseau à relier à la modularité sont les réseaux sans échelles (*scale-free networks*). Un réseau small world est un réseau dont le nombre de noeuds de degré k suit une loi de puissance. Les réseaux sans échelles ont notamment été étudiés par Barabasi lors de l'étude du world wide web, [2]. Ces réseaux sont "ultra-modulaires", ils ont un coefficients de clustering élevés. Il se caractérisent par la présence de sous-graphes ultra-connectés, mais connectés par un *hub*, un noeud connectant d'autres noeuds moins connectés. Par leur distribution de connexions, on retrouve également dans ces réseaux une propriété d'auto-similarité : un sous-graphe autour d'un hub présentera une structure similaire à un sous-graphe plus grand, centré sur un plus gros hub, les hubs étant reliés entre eux. Par cette modularité, ils présentent donc aussi une caractéristique de petit-monde. Ces réseaux sont répandus au sein des structures sociales, par exemple les réseaux de citations entre articles scientifiques, les réseaux sociaux...

Conclusion

[7] : différencie réseaux small world et réseaux hiérarchique modulaires, et statue qu'un réseau peut être "hiérarchique modulaire" même s'il n'est pas small world, mais possède une dimension topologique finie. [12] : statue que un réseau modulaire est small world,

En bref : - techniques de détection de communautés sur des graphes pour détecter une modularité. - Certaines structures se distinguent parmi les graphes modulaires : small-world, scale-free, hiérarchique. ces structures se recoupent. - Au niveau du cerveau, on n'est pas hyper d'accord sur le type de modules, mais les exps montrent une structure. - Echelles des modularités, hyper important. - Réseaux sociaux présentent aussi ces formes de modularité.

1.2.2 Modularité fonctionnelle

La frontière est floue entre modularité structurelle et fonctionnelle. Les deux vont probablement de pair, mais parfois un système ne présente pas de structure de graphe visible, et l'étude de sa connectivité n'est pas possible. L'étude de ce système passe alors par les fonctions qu'il met en jeu : alors, une partie du système répondant à une fonctionnalité particulière sera considérée comme un module du système.

1.2.3 Modularité temporelle

Du point de vue du système dynamique, l'aspect temporel n'est finalement qu'une dimension de plus dans la modularité du système. Les règles locales faisant évoluer le système dynamique amènent à des comportements complexes temporellement parlant. Par exemple, un attracteur fractal est le résultat des trajectoires d'un système dynamique complexe à l'état de chaos.

La modularité s'associe aux séquences dans le cerveau. Les mémoires, l'interaction entre échelles de fonctionnement apportent une modularité supplémentaire.

1.2.4 Modularité émergente ou modularité définie

1.2.5 Emergence d'une modularité et auto-organisation

Est ce que l'auto-organisation d'un système dénote d'une modularité ?

1.3 La modularité, répandue dans les système biologiques

1.3.1 Le cerveau, réseau modulaire fondamental

Un neurone est un système dynamique, donc l'activité électrique est régie par des équation d'évolution dépendant des connexions qu'il reçoit. Le cerveau dans son ensemble est alors, fondamentalement, un agrégat de neurones. 300 neurones dans le ver microscopique *Caenorhabditis elegans*, un million dans le cerveau des insectes, jusqu'à 86 milliards de neurones dans un cerveau humain. Tous ces pics électriques et chimiques nous permettent une prise de décision, une mémoire, de la réflexion, des représentations Un ensemble de capacités que l'on nomme intelligence. Cette intelligence n'est pas localisée dans le cerveau, de ce qu'on sait. Elle résulte de l'activité globale de l'ensemble de neurone, et est ainsi une propriété émergente.

Cet amas obscur est le point de départ des études autour de l'intelligence artificielle. Si on veut créer un systèmes ayant une intelligence émergente, il nous faut comprendre les mécanismes de cette émergence, et donc comprendre les systèmes existant ayant cette capacité. Les réseaux de neurones se sont donc d'abord inspirés du cerveau avant d'être développés plutôt sur un aspect computationnel avec moins de vraisemblance biologique. Cette inspiration biologique ne se limite pas à assembler des neurones : l'architecture cérébrale à plus grande échelle est aussi une source d'inspiration dans la recherche de systèmes intelligents, autrement dite, intelligence artificielle. Et le cerveau semble avoir une architecture particulière : il est modulaire, à plusieurs échelles. Les premières propositions de modèle du cerveau humain datent du début du XXème siècle. Déjà, un découpage en aires est proposé pour expliquer le fonctionnement de cet organe, notamment avec les travaux de Broca et Wernicke qui mettent en lumière des zones du cerveau qui semblent responsable du langage. Le modèle connexionniste du cerveau, formalisé à partir de ces travaux par Geschwind dans les années 60, décompose ainsi le langage en plusieurs fonctions : la compréhension, la lecture et l'action de parler. Ce modèle n'est plus vraiment utilisé, mais l'idée de décomposition en modules reste valable. Avec l'avènement des outils d'imagerie, le cerveau a pu être cartographié plus précisément en un ensemble d'aires, agissant comme modules fonctionnels au sein d'une structure complexe ; ces aires sont elles mêmes composées de modules distincts.

L'étude des aires et des connexions entre zones du cerveau, comme [6] dans le cas du cortex visuel du primate, découpent les zones activées pour la vision en modules distincts, et montrent que ces modules sont connectés. Ces connexions, en fonction des modules, sont réciproques ou non. Les "pathways" du cerveau désignent des modules fortement connectés. Leur détection expérimentale est réalisée en relevant des indicateurs de dépendance (corrélation, ..., en fonction des

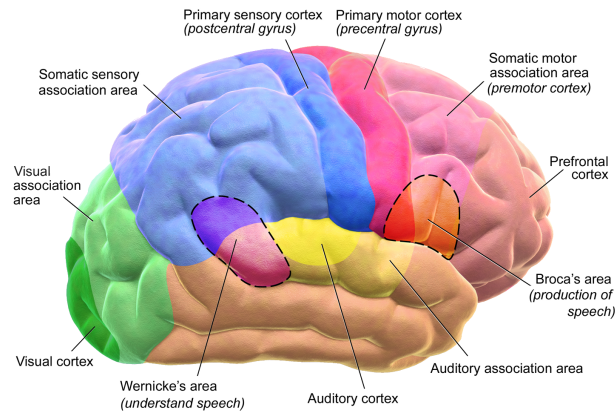


FIGURE 1.2 – Aires du cerveau humain

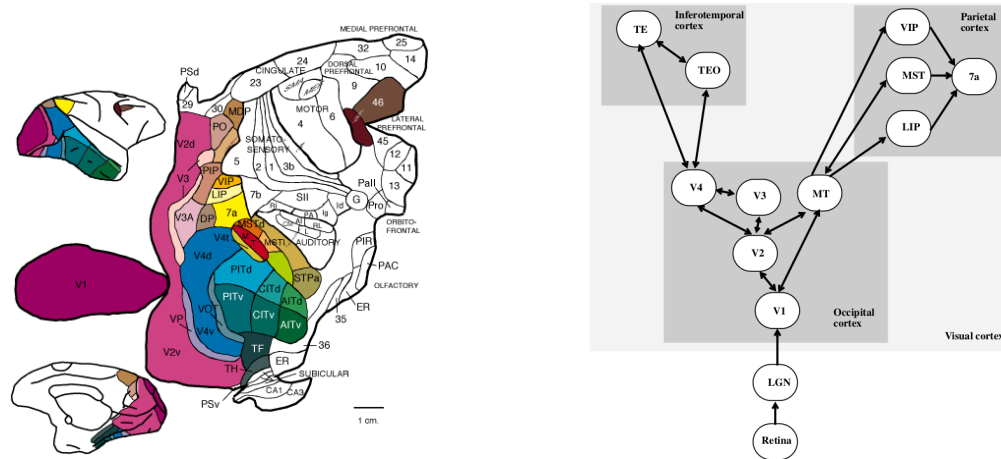


FIGURE 1.3 – carte aplanée des aires du cortex visuel du macaque [6], à gauche, et pathways dans ce cortex visuel entre les aires à droite [16]. L'architecture neuronale présente des rétroaction entre zones, dans une architecture non-hiérarchique.

méthodes utilisées). Ainsi, [16] précise la structure des différentes aires cérébrales en présentant les connexions au sein de ces aires, par exemple la structure présentée en figure 1.3.1 pour l'aire visuelle. Les connexions du cerveau sont présentes à différentes échelles : des pathways existent ainsi au sein de l'aire visuelle du cerveau, mais des boucles de rétroaction entre zones cérébrales sont présentes à plus grande échelle, par exemple la boucle baso-thalamo-corticale. Ces différentes échelles au sein du cerveau amènent [12] à le présenter comme un réseau "hiérarchique modulaire". Cette activité "sans échelle" pourrait notamment permettre au cerveau d'appréhender les notions temporelles, transposant ainsi un réseau spatial en une activité temporelle [3].

La structure cérébrale est donc bien particulière. Aussi, l'inspiration cérébrale des modèles d'intelligence artificielle ne se résume pas au modèle neuronal, les éléments architecturaux et de connexions sont donc à considérer.

1.3.2 Des réseaux et des modules dans tous les systèmes biologiques

Les phénomènes émergents ne se limitent pas au cerveau : tous les systèmes biologiques semblent présenter des fonctionnement issus de systèmes complexes, menant à une régulation

globale d'une quantité, à un comportement commun, autrement dit, l'émergence d'un comportement. On peut par exemple citer l'exemple des nuées d'oiseaux : des centaines ou des milliers d'oiseaux peuvent voler ensemble, sans se heurter ni se disperser, le tout sans avoir d'instructions globale. Cette capacité à rester en nuée émerge des règles locales que connaissent chacun des oiseaux. Ces règles régissent le comportement individuel à avoir avec leur voisins proches. Ces règles simples permettent à ces milliers d'individus de voler en groupe et de se déplacer. Les arbres en forêt s'adaptent, pour pousser sans se toucher, et partageraient même des information en communiquant. Sans forcément parler d'intelligence, on peut tout à fait voir en ces comportements des mécanismes émergents. Le blob,

1.4 Intérêt computationnel des réseaux modulaires

Barabasi : Hypothèse que les réseaux small world présentent un avantage evolutionnaire. Modularité hiérarchique présente l'avantage de maintenir une activité dans le réseau sans que ça ne colonise tout ni ne s'éteigne, ce qui est nécessaire pour la computation.

1.4.1 Modularité, complexité et émergence d'un apprentissage ?

Systèmes complexes et emergence : possibilité d'apprentissage, exemple du reservoir computing

La modularité est liée a la complexité des systèmes, donc l'emergence de comportements chaotiques et/ou synchronisés.

SYSTEMATIC GENERALIZATION : WHAT IS REQUIRED AND CAN IT BE LEARNED ? : Our findings show that the generalization of modular models is much more systematic and that it is highly sensitive to the module layout, i.e. to how exactly the modules are connected.

Simplicité de la modularité : exemple de construction des fractales, exemple du rigaudon

Parler des architectures de Hebb (précurseur) ici ou avant ?

réponse a un problème de contrainte physiques, énergétiques

- Parallelisme, calcul et small world networks : réponse a un problème de contrainte physiques, énergétiques. - Calcul distribué - Automates cellulaires ?

- Exemple des puces neuromorphiques, calcul embarqué

Conclusion : En décentralisant le calcul, on cherche aussi à plus facilement l'utiliser via des architectures neuromorphiques.

1.4.2 Types de connexions

Opaque vs tout savoir

1.5 Modularité dans les réseaux de neurones

Au sein des systèmes complexes, les réseaux de neurones artificiels se sont directement développés en s'inspirant des réseaux biologiques. Hebb formule ainsi en 1949 "neurons that fire together, wire together". Les réseaux de neurones développés depuis sont alors inspirés de la biologie, en s'en éloignant plus ou moins. Quel que soit leur éloignement, ce sont des systèmes complexes dans le sens où il s'agit d'un grand nombre de neurones reliés entre eux, et qu'on développe dans le but de l'émergence d'une propriété d'apprentissage et de généralisation.

Ces réseaux sont par exemple le perceptron, les cartes auto-organisatrices. L'intérêt de chercher des aspect modulaires des réseaux de neurones a été formulé par exemple par [18] :

A partir du perceptron, les réseaux de neurones profond ont été développés comme une assemblée de perceptrons reliés hiérarchiquement (fin 90)

1.5.1 Deep Learning

Boîtes noires qui ont des performances remarquables sur tous les domaines, leur représentation et compréhension est quant à elle toujours un challenge. Ajouter un aspect modulaire non-hiérarchique dans les calculs de ce genre de réseau, pour augmenter la vitesse d'apprentissage ou encore pour

Réseaux de neurones profond et modulaires

- Réseaux qui apprennent à s'organiser en modules. Interet. Limites? Performances? [1, 10]
 "The NMN approach is intuitively appealing but its widespread adoption has been hindered by the large amount of domain knowledge that is required to decide (Andreas et al., 2016) or predict (Johnson et al., 2017; Hu et al., 2017) how the modules should be created (parametrization) and how they should be connected (layout) based on a natural language utterance. Besides, their performance has often been matched by more traditional neural models" (systematic generalization article)

Utiliser des modules pour mieux représenter les réseaux de neurones profond

- Trouver des modules dans les réseaux pour les expliquer? [17, 5] are neural net modular : "it uses different modules for very different functions = Pspecialize," et "it uses the same module for identical functions that may have to be performed multiple times = Preuse" - Reconciling deep learning with symbolic artificial intelligence : representing objects and relations(2019) Pb du deep learning = Data inefficiency (comparé à l'humain) ;Poor generalisation ; Lack of interpretability.

1.5.2 Réseaux auto-organisés

Les réseaux auto-organisés sont directement inspirés des réseaux biologiques. Au sein de ces réseaux Plus qu'en deep learning, les réseaux de neurones auto-organisés - Auto-organisation prend une profonde inspiration biologique, tout comme les modules. - Exemple de réseaux auto-organisés modulaires : développer dans la partie suivante. [11]

[15]

Lefort

Travaux préliminaires de l'équipe -> Bassem, Ménard?

1.5.3 Construire une architecture modulaire

A mettre dedans :

- Réseaux top down / modulaires? définition, à quel point un réseau est modulaire, qu'est ce qu'on appelle réseau modulaire? Fonction définies au préalable vs émergence des fonctions. Modules définis au préalable vs émergence des modules.

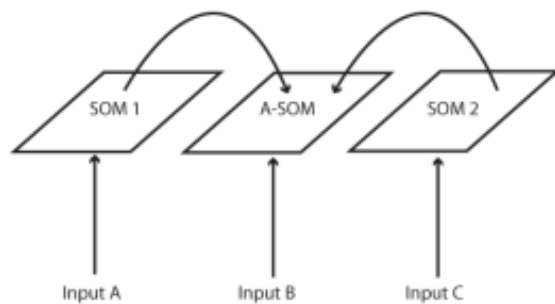


FIGURE 1.4 – A-SOM model [9]

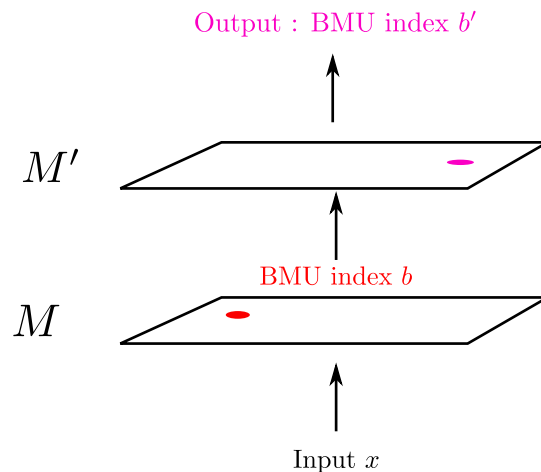


FIGURE 1.5 – HSOM model [11]

1.6 Enjeux d'une architecture modulaire de SOMs

On connaît plutôt bien les SOM, mais on sait que des comportements nouveaux peuvent émerger lorsqu'on met en interaction des systèmes étudiés séparément. On peut donc se poser la question des comportements qui peuvent survenir dans ce cas.

Dans les structures de cartes étudiées, modularité forte dans le sens où les fonctions des modules sont prédéfinies. Si on ne fournit que les règles d'interaction, où est-ce qu'on se situe ?

Position du manuscrit : architecture

Plan de la partie !

1. Intro : Notre monde est modulaire, en tout cas nous l'interprétons en tant que tel. Proposé déjà dans les années 60 - brève histoire des réseaux de neurones ? Remarquer que nous, observateur, raisonne et se construit modulairement. Il nous est difficile de concevoir les choses autrement. Les disciplines scientifiques par ex, un domaine obéit à des principes
2. Qu'appelle-t-on la modularité ? Définitions claires et propriétés
 - Définition
 - (a) Structurelle, dans les systèmes réseaux
 - (b) Fonctionnelle, dans les systèmes dont on ne connaît pas la structure ?
 - (c) Temporelle / mais est-ce que le temps ce n'est qu'une dim de plus
 - (d) Modularité hiérarchique attention : deux defs à hiérarchiques. Soit c'est une histoire de connexions, soit d'auto-similarité. On parle nous de l'auto similarité!!! Au contraire l'autre forme de hiérarchie n'est pas observée en bio.
 - Propriétés de la modularité
 - (a) Auto-organisation
 - (b) Types de connexions
 - (c) Emergence
3. Exemples Biologiques et computationnels
 - (a) Biologie : opti evolution a priori...

- (b) Computationnel : pareil.
- 4. Maintenant qu'on sait ce que sont les systèmes modulaires, en quoi on peut faire de l'apprentissage avec ? Quels sont les intérêts ?
 - (a) Exemples de réseaux auto-organisés
 - (b) Exemples de réseaux de neurones : deep learning

Chapitre 2

Un réseau de neurones auto-organisé : les cartes de kohonen

// Kohonen : il faut surprendre encore ! Par quel bout le prendre ? → Appuyer sur les cartes 1D → Comment ça se fait qu'on les utilise pas de ouf ? → Intérêt de la topologie de la carte. Dans une carte seule, est ce que c'est vraiment utile ? → Questionnement informatique : qu'est ce qui se passe en fait dedans, mais c'est quand même rigolo.

2.1 Principe Général

Une carte de Kohonen est un graphe dans lequel chaque noeud possède un poids ω appartenant à l'espace des entrées. L'algorithme repose ensuite sur l'adaptation de ces poids, en prenant en compte les connexions dans le graphe, afin de représenter les données d'entrées. Ainsi, n'importe quel graphe pourrait être considéré ; le plus souvent, une grille 2D est utilisée.

2.2 Approche topologique des cartes de Kohonen

La notion de voisinage et de topologie est un élément clé des cartes de Kohonen. Le voisinage est en effet pris en compte lors de l'apprentissage et lors de l'interprétation des cartes. Cependant, ce voisinage est généralement défini, dans les applications des cartes, comme un bonus par rapport aux KMeans, une aide à la convergence et à la vitesse de dépliage. Pourtant c'est la l'essence même d'une carte de Kohonen : projeter des éléments sur un graphe, ce qui nous permet de faire des calculs sur des positions plutôt que des données de grandes dimensions.

2.3 Principes d'organisation

2.4 Travaux préliminaires sur les architectures de cartes de Kohonen

2.5 A trier

De la biologie à la computation : patterns temporels des neurones impulsifs vs SOM.

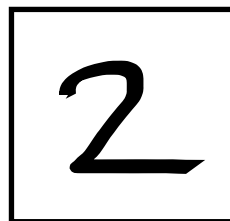
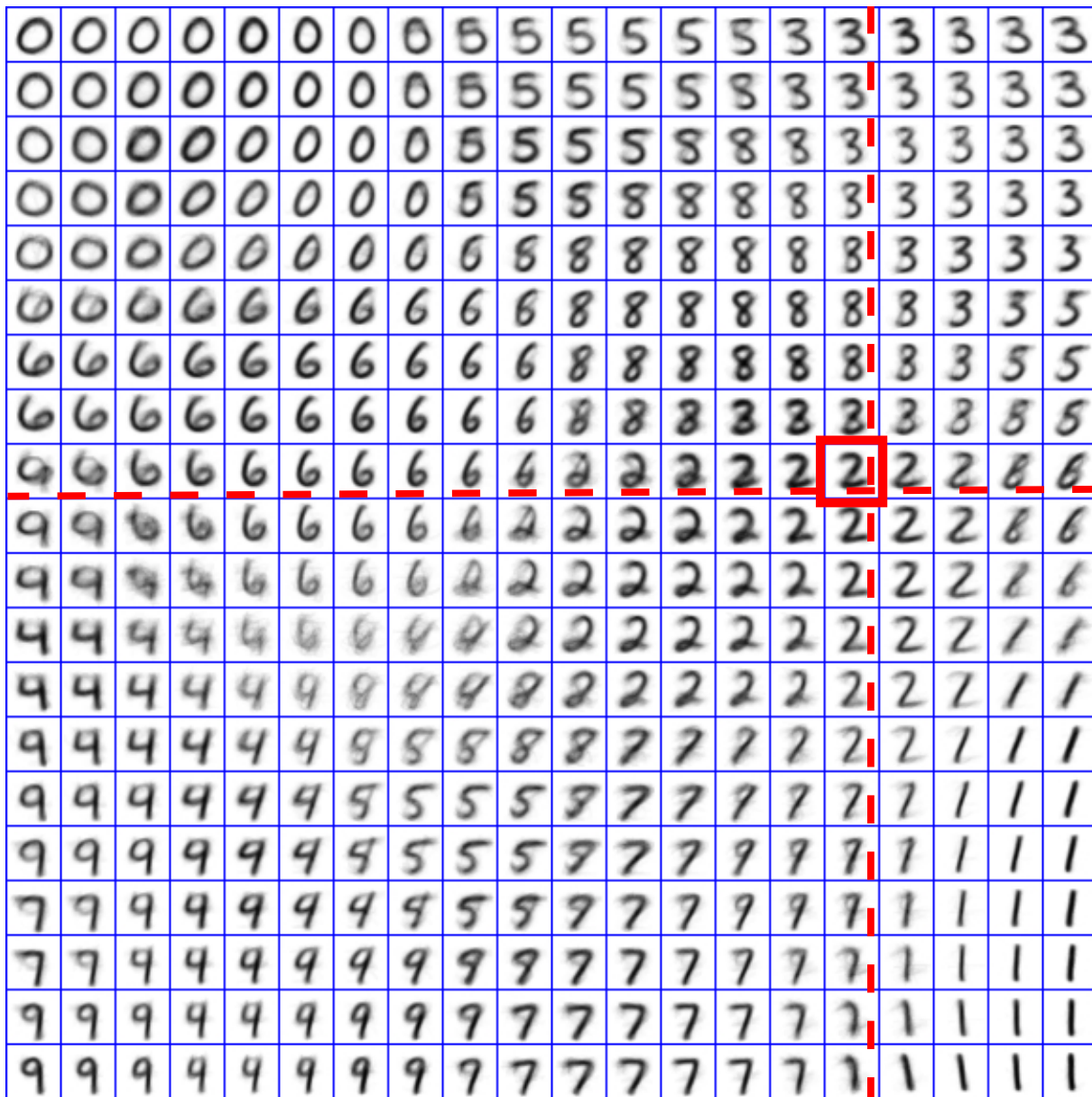


FIGURE 2.1 – Une carte de Kohonen s’organise en zones dont les poids sont proches dans l’espace des entrées. Chaque entrée présentée à la carte peut alors être représentée par la valeur de la position du BMU correspondant dans la carte. Les entrées sont projetées sur le carré $[0, 1] \times [0, 1]$.

Chapitre 3

Construction d'une architecture modulaire de cartes auto-organisatrices

Le but de cette thèse est de proposer un modèle permettant d'associer des cartes auto-organisatrices dans n'importe quel type d'architecture, comme une sorte de brique de base. En particulier, on cherchera à construire des architectures non-hiérarchiques de cartes, exemple en figure 3.1. Nous nous plaçons donc dans le cadre de modules pré-établis, dont les entrées ont été connectées par avance. Les poids de chaque carte seront quant à eux appris, avec comme objectif que les cartes apprennent leurs entrées mais puissent également distinguer un état global de l'architecture. Pour les entrées, nous nous plaçons dans un cadre de multi-modalité, détaillé au chapitre suivant. Les différentes cartes prendront des données d'entrées sur différents espaces.

3.1 Description de l'algorithme

3.1.1 Carte de Kohonen classique

Rappelons les notations concernant une carte de Kohonen standard. Prenons un ensemble de données d'entrées, dans lequel chaque élément est un vecteur d'un espace D . On a défini une distance d sur D , généralement la distance euclidienne. La carte de Kohonen construite sur ces entrées est un graphe, généralement une ligne 1D ou une grille 2D de N noeuds. Chaque noeud possède un poids associé $\omega_e \in D$ ou *prototype*, du même espace que les entrées. et une *position* i dans la carte. Ces positions sont ensuite indexées entre 0 et 1 par $p = \frac{i}{N}$ pour l'homogénéité des

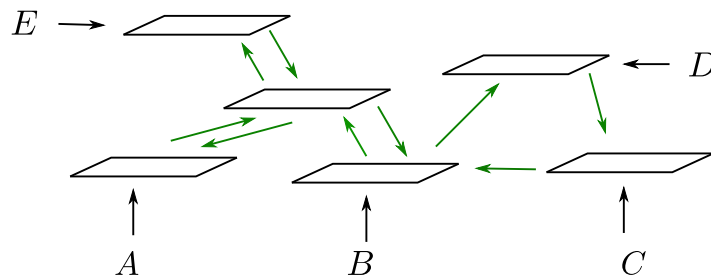


FIGURE 3.1 – Exemple d'architecture modulaire *non-hiérarchique* de cartes de Kohonen. Les entrées sont A, B, C, D, E quelconques. Chaque carte peut ou non prendre une entrée ; les connexions sont réciproques ou non.

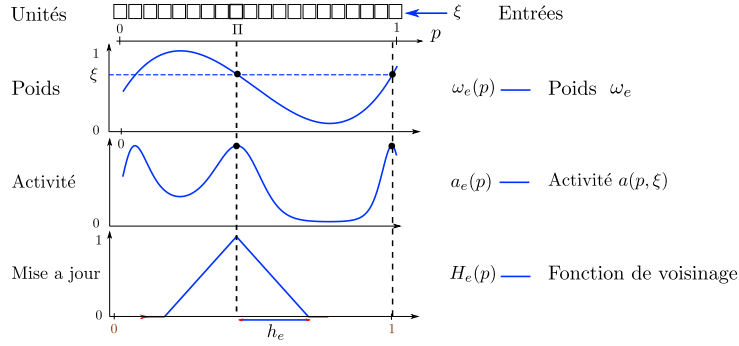


FIGURE 3.2 – Notations utilisées dans une carte de Kohonen simple

calculs. L'ensemble des poids est noté $\omega_e(p), p \in [0, 1]$. L'algorithme se décompose de la façon suivante :

1. Une entrée ξ_t est présentée à la carte.
2. L'unité ayant le poids le plus proche de ξ_t selon la distance d est choisie comme *Best Matching Unit* de la carte. Sa position est notée Π .
3. Chaque poids ω_e est déplacé vers l'entrées ξ , en fonction de sa distance dans la carte à la best matching unit :

$$\omega_e(p, t + 1) = \omega_e(p, t) + \alpha h(\Pi, p)(\xi - \omega_e(p, t)) \quad (3.1)$$

$h(\Pi, p)$ est la *fonction de voisinage*. Elle est maximale en $p = \Pi$ et décroissante autour de cette position. Dans notre étude, les fonctions de voisinage sont triangulaire, donc maximale en Π , décroissante sur le rayon de voisinage h_e et nulle après.

Lors de l'étape 2 de l'algorithme, une activité peut être calculée, au lieu d'une distance pour choisir le BMU. Ce dernier est alors choisi comme $\Pi = \arg \max_p (a(\xi, p))$. Nous utiliserons cette solutions dans notre modèle. Les notations au sein d'une carte sont résumées en figure 3.2.

3.1.2 Modèle : CxSOM

Décrivons maintenant le modèle CxSOM étudié dans cette thèse. Dans ce modèle, l'algorithme original de Kohonen est modifié afin de connecter des cartes entre elles, et d'autoriser des connexions non-hiérarchiques. Définissons la connexion entre deux cartes. Une carte A est connectée à une carte B lorsque la carte B prend en entrée la position du BMU de la carte A. Considérons G , le graphe de connexions des cartes. Ce graphe est *orienté* et les *boucles* sont autorisées. C'est ce qu'on appellera *architecture non-hiérarchique* de cartes, par opposition à des architectures comme HSOM dans laquelle le BMU d'une carte A nourrit une carte B de façon unidirectionnelle. Chaque carte aura ainsi plusieurs entrées : une entrée *externes* dans un espace d'entrée, facultative, et k entrées *contextuelles* qui sont les positions des BMU des cartes qui lui sont connectées. Par ailleurs, la recherche du BMU doit être modifiée par rapport à l'originale : les rétroactions entre les cartes sont autorisées, la position du BMU de la carte A va donc influencer la position du BMU de la carte B, lequel modifie à nouveau le BMU de la carte A, etc. Notre algorithme présente donc deux modifications principales :

- Les cartes possèdent plusieurs entrées, externes et contextuelles. Le calcul de l'activité est donc modifié afin de prendre en compte ces différentes couches d'entrées.

— La recherche du BMU est modifiée afin de gérer les rétroactions entre cartes.

La description du modèle CxSOM est détaillée en figure 3.4, dans un cas où une carte reçoit deux connexions, et l'algorithme explicité en ??.

Gestion des entrées externes et contextuelles

A un pas d'apprentissage t , une carte M reçoit en entrée une entrée *externe* notée ξ_t et K entrées *contextuelles* notées $\gamma_{0t}, \dots, \gamma_{Kt}$, qui sont les BMU Π des cartes qui lui sont connectées. La carte possède donc $k + 1$ couches de poids. ω_e correspond à l'entrée externe et $\omega_{e0}, \dots, \omega_{eK}$ aux entrées contextuelles. On calcule une activité séparément sur chaque couche de poids selon la formule suivante :

$$a(p, x) = \exp\left(-\frac{(\omega(p) - x)^2}{2\sigma^2}\right) \quad x = \xi_t \text{ ou } \gamma_{kt}, \quad \omega = \omega_e \text{ ou } \omega_{ek} \quad (3.2)$$

Les activités contextuelles sont moyennées en une activité $a_c(p, \gamma_t)$, avec $\gamma_t = (\gamma_{0t}, \dots, \gamma_{Kt})$. Les activités externes et contextuelles sont enfin fusionnées en une activité globale :

$$a_g(p, \xi_t, \gamma_t) = \sqrt{a_e(p, \xi_t)(\beta a_e(p, \xi_t) + (1 - \beta)a_c(p, \gamma_t))} \quad (3.3)$$

Une convolution est appliquée sur cette activité globale. Cela évite les effets de plateau. Cette activation globale est utilisée pour déterminer le BMU de la carte.

Gestion des rétroactions dans l'architecture

Contrairement à une carte simple, on ne peut pas calculer tous les BMUs de l'architecture en prenant l'argmax de a_g dans chaque carte. A cause des influences mutuelles entre cartes, calculer le BMU d'une des cartes modifie les entrées des autres cartes de l'architecture, et donc leur BMU. Cette recherche est donc réalisée par un processus dynamique que l'on appellera *relaxation*, menant à un consensus entre cartes : on cherche le point, s'il existe, où chaque BMU maximise l'activité globale de chaque carte.

Le processus de relaxation est donc une boucle imbriquée dans un pas d'apprentissage de l'architecture, indexée par τ . Notons $\Pi[i]$ la position du BMU de la carte i , et $\mathbf{\Pi} = (\Pi[0], \dots, \Pi[n])$, avec n le nombre de cartes de l'architecture. Au début d'un pas d'apprentissage, chaque carte est nourrie avec une entrée externe ξ_t^i , et les activités externes $a_e^i(\xi_t^i, p)$ de chaque carte peuvent être calculées. La recherche du BMU suit donc le processus de relaxation suivant :

1. Dans chaque carte i , la position Π^i est initialisée à $\arg \max_p(a_e^i(\xi_t^i, p))$. Les entrées contextuelles sont alors initialisées en prenant le BMU correspondant aux connexions de l'architecture.
2. Tant que toutes les positions Π^i ne sont pas stables,
 - (a) Dans chaque carte i , calculer les activités contextuelles et globales, définissant ainsi $p^{*i} = \arg \max_p(a_g(p, \gamma^i, \xi^i))$
 - (b) Déplacer Π^i vers p^{*i} : $\Pi^i \leftarrow \Pi^i \pm \Delta$ si $|\Pi^i - p^{*i}| \geq \Delta$, $\Pi^i \leftarrow p^{*i}$ sinon
3. Le BMU de chaque carte est pris comme la valeur finale stable de ce processus dynamique. Cette valeur est utilisée pour la mise à jour des poids.

Il peut arriver que les positions se stabilisent sur un cycle limite. Dans ce cas, on arrêtera la relaxation arbitrairement ; ce phénomène étant ponctuel, il n'influencera pas l'apprentissage. Les paramètres des cartes de l'architecture sont choisis pour éviter de telles situations.

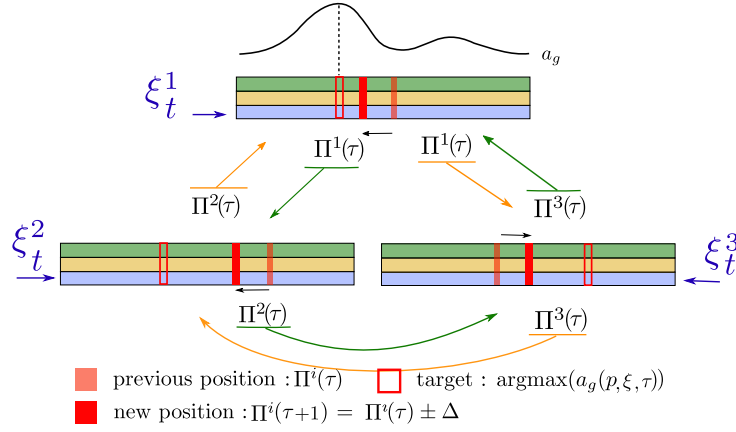


FIGURE 3.3 – description d’une étape de la relaxation dans l’architecture, aboutissant à un consensus entre cartes. Au sein d’une même itération t , les position des BMU Π sont légèrement déplacées jusqu’à ce que toutes les positions Π des cartes de l’architecture soient stable. Ces positions maximisent collectivement les activités globales de chaque carte.

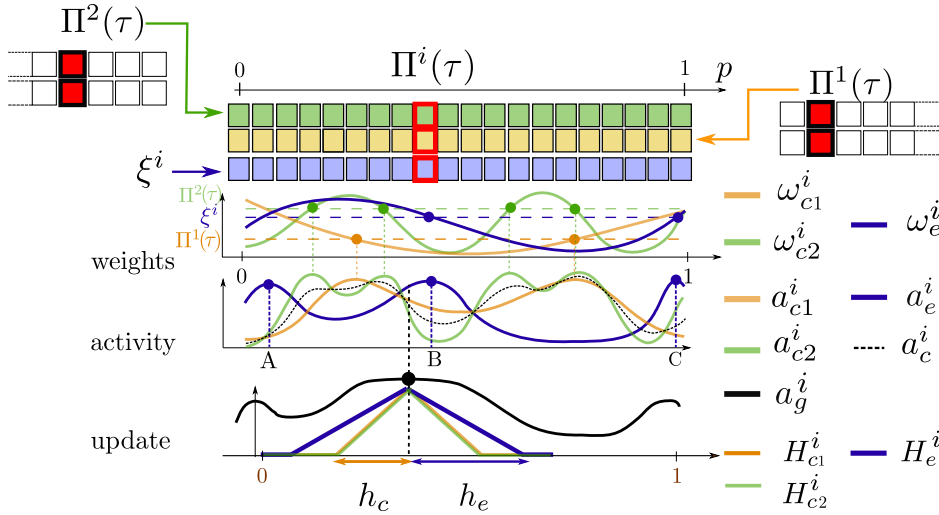


FIGURE 3.4 – Description d’une carte au sein d’une architecture CxSOM. La carte reçoit deux connexions de cartes voisines, et possède donc deux couches contextuelles

Mise à jour des poids

Les poids sont mis à jour par rapport à leurs entrées respectives suivant l'équation 3.1. Le BMU d'une carte est ainsi commun à toutes les couches. Les rayons de voisinage h_e et h_c ont des valeurs différentes ; celles-ci seront détaillée en partie suivante.

Tests

Les expériences faites sur l'architecture se décomposent en une phases d'apprentissage et phases de test. Pendant les tests, la mise à jour des poids des cartes est gelée et seuls le calcul des activités et le processus dynamique de sélection du BMU sont effectués.

Récapitulatif

3.2 Choix des paramètres

3.2.1 Influence des rayons de voisinage

3.2.2 Influence des autres paramètres

3.2.3 Compatibilité en 2D

3.3 Analyse de la relaxation

L'apprentissage conjoint des cartes repose sur la relaxation au sein d'une itération. On cherche donc à vérifier si la relaxation converge vers une valeur quelle que soit l'entrée, et si elle est pertinente en large dimension avec de nombreuses cartes.

3.3.1 Analyse expérimentale

3.3.2 Champs de BMU

3.3.3 Limitations et possibilités en grande dimension

3.4 Implémentation

L'implémentation des expériences a été réalisée via l'environnement CxSOM [?].

3.5 Perspectives d'évolutions

Chapitre 4

Analyser l'organisation : une approche par variables aléatoires

Avant de présenter les performances d'un algorithme, il s'agit de définir plus précisément ce qu'on attend de ce système et comment le représenter. L'architecture CxSOM se présente comme une construction qui répond à un questionnement structurel des réseaux de neurones. Mais au juste, qu'attend t-on de ce réseau de neurones ? De la prédiction, de l'organisation ? Les cartes de Kohonen sont habituellement utilisées dans un objectif de clustering, ou associées à d'autres algorithmes de prédiction utilisant leurs propriétés structurelles. En étude préliminaire pour CxSOM, il s'agit de comprendre le comportement de l'architecture de cartes.

4.1 Cas d'utilisation : les entrées multimodales

4.1.1 Définition et inspiration biologique

4.1.2 Formalisme

4.1.3 Perspectives

Le formalisme présenté, avec des entrées multimodale comme fonction de variable cachées n'est pas forcément général.

4.2 Représentation des entrées

4.3 Information apprise par une carte

Une idée est de déterminer si une carte a gagné de l'information sur le modèle générant les entrées. Dans le cas simple, ce modèle peut être entièrement représenté par U ; chaque carte peut être représentée par son BMU, considéré comme la seule sortie de la carte. En traçant U en fonction de Π , le BMU d'une carte, on observe directement si une carte a été capable de lever l'ambiguïté sur le modèle en distinguant les entrées selon leur variable cachée U . Cette ambiguïté est levée si U est une fonction de Π . Cette fonction est observée dans le cas des cartes jointes.

Cette propriété, dans le cas 1D, peut être calculée par l'information mutuelle entre U et Π . Plus précisément, par $\frac{I(U,\Pi)}{H(U)}$, avec $H(U)$ l'entropie de U . En effet, dans le meilleurs des cas, U est une fonction parfaite de Π et donc $H(U|\Pi) = 0$: en connaissant Π , on connaît totalement U . Alors, $I(U,\Pi) = H(U) - H(U|\Pi) = H(U)$. Notre indicateur vaut alors 1 lorsque U est

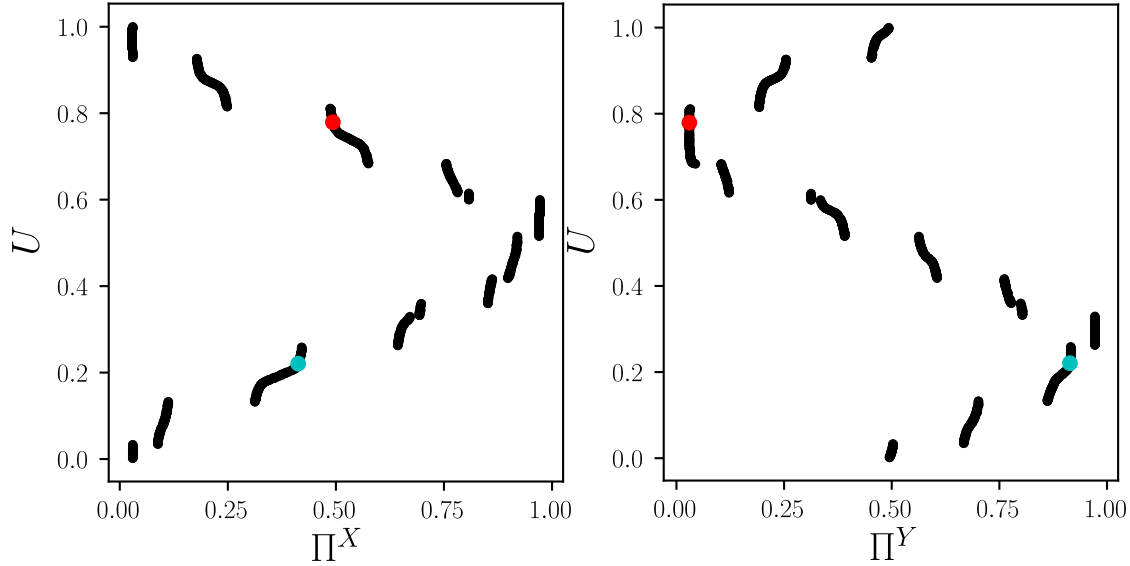


FIGURE 4.1 – Pour l'échantillon de test, valeur de U en fonction des valeurs du BMU Π dans chacune des cartes. On voit que U est une fonction du BMU dans chaque carte, contrairement au cas où les cartes apprendraient indépendamment sur les mêmes entrées, voir figure 4.2.

une fonction parfaite de Π . De plus Π est forcément une fonction de U car l'algorithme est déterministe : à une entrée correspond une sortie, toujours la même, donc $(I(U, \Pi) = H(\Pi))$. Notre indicateur estimant l'information portée par le BMU d'une carte sur la variable cachée du modèle U est donc

$$\frac{H(\Pi)}{H(U)}$$

Cet indicateur doit être estimé en discrétisant les variables, donnant une entropie nécessairement positive et strictement supérieure à 0. L'évolution de l'indicateur au cours de l'apprentissage est donnée en figure 4.3. Cet indicateur est calculé en moyenne pour 100 réalisations de l'apprentissage, avec des poids initiaux différents.

Choses à faire

- Cette valeur est uniquement calculée pour un modèle connu, et en 1 dimension forcément. Peut-on avoir des équivalents en plus de dimension ?
- Il existe des quantités mesurant l'information portée par un symbole sur une variable, une sorte d'info mutuelle locale. On sait que $I(U, \Pi) = H(\Pi)$, et on veut que $I(U, \Pi) = H(U)$, mais comment est-elle répartie entre les BMUs ? Est-ce pertinent de se pencher sur ces quantités ?

4.4 Représenter une carte au sein d'une architecture

Représentation des poids, des entrées, des BMU - analyse

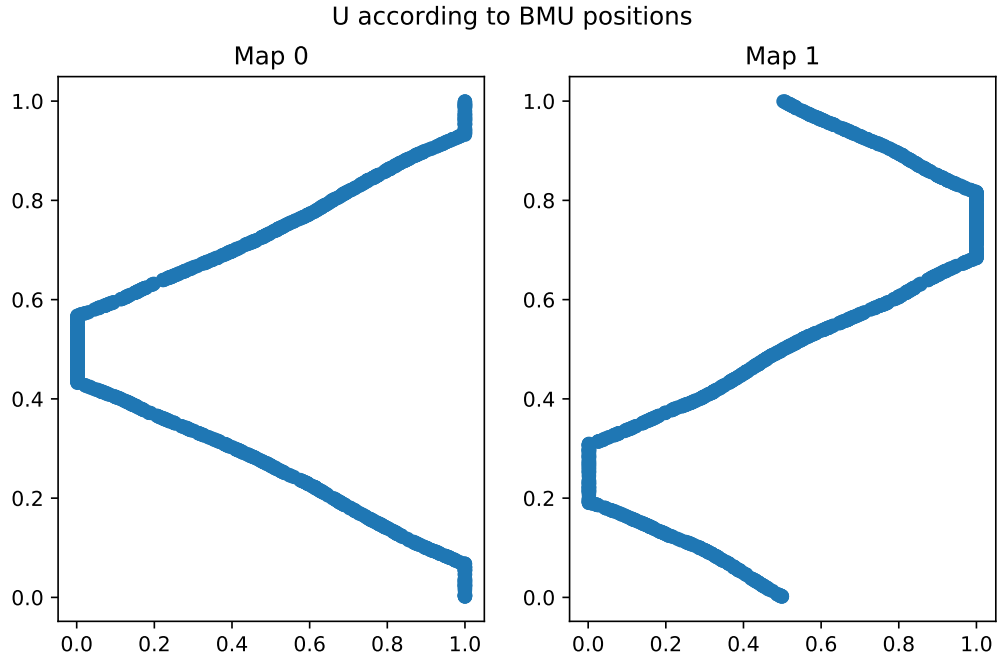


FIGURE 4.2 – Pour l'échantillon de test, entrée sur un cercle, valeur de U en fonction des valeurs du BMU Π dans chacune des cartes, lorsque les cartes M_x et M_y ne sont pas connectée. Chacune des cartes n'a aucune information de plus que celle portée par son entrée sur l'état global du système U , et Π n'est donc pas une fonction de U dans chaque carte.

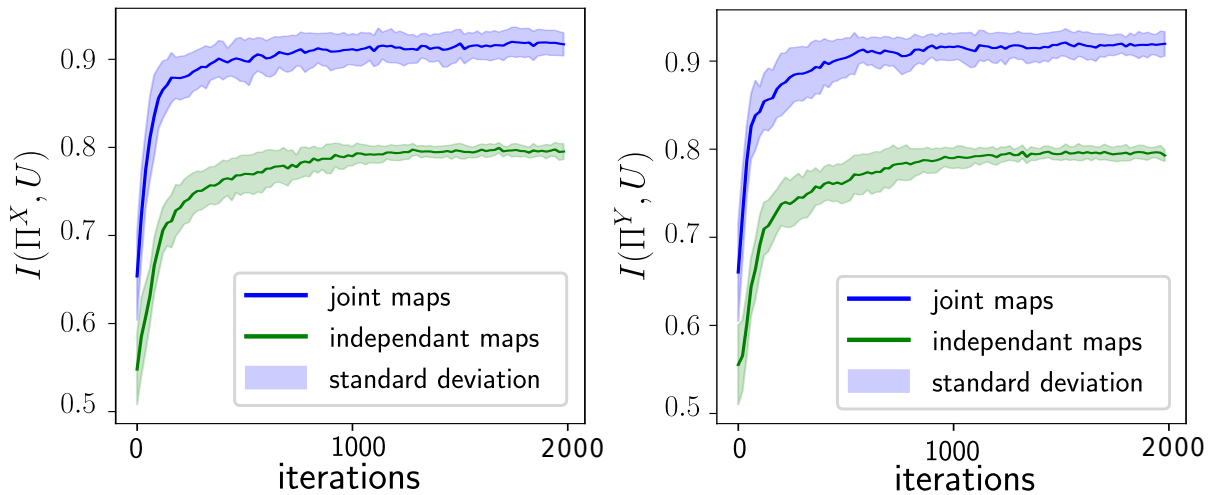


FIGURE 4.3 – Evolution de l'indicateur relatif à l'information mutuelle entre Π et U dans chaque carte au cours de l'apprentissage. Cet indicateur est comparé à celui calculé dans le cas où les cartes apprennent séparément.

4.5 Prédiction d'entrée

Prédction sur des données jouets

Prédiction sur drone

Bien se placer dans le contexte “on va chercher a omprendre ce système dynamique”.

Formaliser le problème en terme de variables aléatoires

Conclusion

Bibliographie

- [1] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and D. Klein. Neural module networks. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 39–48, 2016.
- [2] A. L. Barabasi and Eric Bonabeau. Scale-free networks. *Scientific American*, 288 :60–69, 2003.
- [3] J. He Biyu. Scale-free brain activity : past, present, and future. *Trends in Cognitive Sciences*, 18(9), September 2014.
- [4] Rodney A. Brooks. A robust layered control system for a mobile robot. *IEEE J. Robotics Autom.*, 2 :14–23, 1986.
- [5] R’obert Csord’as, Sjoerd van Steenkiste, and J. Schmidhuber. Are neural nets modular ? inspecting functional modularity through differentiable weight masks. *ArXiv*, abs/2010.02066, 2021.
- [6] Daniel J. Felleman and David C. Van Essen. Distributed hierarchical processing in the primate cerebral cortex. 1991.
- [7] Claus C. Hilgetag and Alexandros Goulas. Is the brain really a small-world network ? *Brain Structure & Function*, 221 :2361 – 2366, 2015.
- [8] Judit Horváth, István Szalai, and Patrick De Kepper. An experimental design method leading to chemical turing patterns. *Science*, 324 :772 – 775, 2009.
- [9] M. Johnsson, C. Balkenius, and G. Hesslow. Associative self-organizing map. In *Proc. IJCCI*, 2009.
- [10] Louis Kirsch, Julius Kunze, and D. Barber. Modular networks : Learning to decompose neural computation. In *NeurIPS*, 2018.
- [11] J. Lampinen and E. Oja. Clustering properties of hierarchical self-organizing maps. *Journal of Mathematical Imaging and Vision*, 1992.
- [12] D. Meunier, R. Lambiotte, and E. Bullmore. Modular and hierarchically modular organization of brain networks. *Frontiers in Neuroscience*, 4, 2010.
- [13] S Milgram. The small world problem. *Psychology today*, 2 :60–67, 1967.
- [14] Harold J. Morowitz. The mind, the brain, and complex adaptive systems. 1995.
- [15] German I. Parisi, Jun Tani, Cornelius Weber, and Stefan Wermter. Lifelong learning of spatiotemporal representations with dual-memory recurrent self-organization. *Frontiers in Neurorobotics*, 2018.
- [16] Edmund T. Rolls and Gustavo Deco. Computational neuroscience of vision. 2002.
- [17] C. Watanabe, Kaoru Hiramatsu, and K. Kashino. Modular representation of layered neural networks. *Neural networks : the official journal of the International Neural Network Society*, 97 :62–73, 2018.

- [18] Stefan Wermter, Jim Austin, David Willshaw, and Mark Elshaw. Towards novel neuroscience-inspired computing. 2001.