

## Chapitre 2 : Fondements Théoriques et Origines du DSL

<b>2.1. Genèse Historique et Inspirations Multidisciplinaires .....</b>	<b>133</b>
2.1.1. Les Premières Pistes d'Auto-Organisation .....	133
2.1.2. SOM, Hopfield et la Génération des Réseaux Associatifs .....	136
2.1.3. Entre Neurosciences Computationnelles et Physique Statistique .....	141
2.1.4. Précurseurs de la Synergie Multimodale .....	146
2.1.5. Origine du DSL et Ses Premiers Manifestes .....	150
<b>2.2. Principes Mathématiques de Base.....</b>	<b>156</b>
2.2.1. Définition des Entités et de la Fonction de Synergie .....	156
2.2.2. Mise à Jour des Pondérations : Formule Générale .....	163
2.2.3. Règles de Parsimonie et Seuil de Connexion .....	168
2.2.4. Notions d'États Internes et Auto-Organisation .....	173
2.2.5. Exemples Illustratifs.....	179
<b>2.3. Hypothèses de Stabilité et Émergence de Clusters.....</b>	<b>186</b>
2.3.1. Point Fixe, Attracteurs : Définitions et Existence .....	186
2.3.2. Attracteurs Multiples et Oscillations.....	194
2.3.3. Analyse de la Formation de Clusters.....	201
2.3.4. Influence du Bruit et des Perturbations .....	207
2.3.5. Limites Théoriques et Questions Non Résolues .....	211
<b>2.4. Raccords avec la Physique Statistique et la Théorie des Systèmes Dynamiques .</b>	<b>218</b>
2.4.1. Systèmes Dynamiques Discrets ou Continus .....	218
2.4.2. Parallèles avec les Modèles de Spin (Ising, Potts, Hopfield).....	225
2.4.3. Notion d'Énergie ou de Fonction Potentielle.....	231
2.4.4. Rôle de l'Inhibition et de la Saturation Synaptique .....	238
2.4.5. Approches Hybrides et Collaboration Interdisciplinaire .....	243
<b>2.5. Perspectives Historiques et Liens avec l'IA Moderne .....</b>	<b>249</b>
2.5.1. Évolutions Récentes et Convergence avec le Deep Learning.....	249
2.5.2. Extensions Symboliques et Neuro-Symboliques .....	252
2.5.3. Apport en Robotique et Contrôle Adaptatif.....	258

<b>2.5.4. Comparaison avec d'Autres Paradigmes (RL, GNN, etc.).....</b>	<b>263</b>
<b>2.5.5. Transition vers les Chapitres Suivants .....</b>	<b>266</b>

### 2.1. Genèse Historique et Inspirations Multidisciplinaires

Le **DSL** n'est pas né ex nihilo. Il s'inscrit dans une **trajectoire historique** plus large, marquée par les travaux en **cybernétique**, la recherche sur la **plasticité neuronale** et les premières tentatives de construire des systèmes capables de s'organiser sans supervision stricte. L'objectif de ce sous-chapitre (2.1) est de dégager les **facteurs historiques** et les **influences** majeures qui ont contribué à l'émergence progressive de la notion d'auto-organisation, pour mieux comprendre le contexte dont le DSL se réclame.

#### 2.1.1. Les Premières Pistes d'Auto-Organisation

Les premières tentatives pour expliquer comment un ensemble d'éléments (neurones, agents, mécanismes cybernétiques) peut parvenir à une **organisation** cohérente sans chef d'orchestre remontent aux années 1940-1960. Elles se sont nourries des hypothèses issues de la **cybernétique** et des premiers constats sur la **plasticité synaptique** dans le cerveau. Les sections qui suivent (2.1.1.1 et 2.1.1.2) se concentrent sur cette phase pionnière, préalable essentiel à l'éclosion de l'apprentissage auto-organisé.

##### 2.1.1.1. Rôle de la Cybernétique et des Premiers Travaux sur la Plasticité Neuronale

Dans les années 1940, la **cybernétique**, introduite par **Norbert Wiener**, s'est concentrée sur l'étude des **boucles de rétroaction** présentes dans les systèmes vivants ou mécaniques. L'enjeu consistait à comprendre la manière dont un organisme, ou un dispositif artificiel, parvient à **réguler** son comportement en réagissant aux informations provenant de l'environnement. Dans ce cadre, plusieurs chercheurs ont rapidement émis l'hypothèse qu'un **réseau** de neurones ou de capteurs, guidé par des règles locales, pourrait atteindre une forme d'**organisation interne** sans exiger de commande globale.

Au sein de cette vision, la perspective de **Norbert Wiener** a mis en évidence que la rétroaction (feedback) pouvait autant stabiliser qu'instabiliser un système, selon la nature et l'orientation de la boucle. Cette découverte a suscité l'idée de concevoir un **réseau adaptatif**, entendu comme un ensemble d'unités  $x_i$  soumises à des liaisons  $\omega_{i,j}$ . La question se posait alors : si chaque liaison s'ajuste localement, pourrait-il se produire un ordre global émergent ? Cette interrogation s'inscrit dans le droit fil des principes de la **cybernétique** : un réseau pourrait, de façon autonome, moduler ses connexions pour converger vers un comportement collectif stable ou cohérent, sans qu'une autorité centrale vienne en spécifier toutes les interactions.

En parallèle, dans les années 1950 et 1960, les premiers **travaux** sur la **plasticité neuronale** se sont développés. Des expériences en neurosciences laissaient supposer que les **connexions synaptiques** – c'est-à-dire les poids entre neurones – se modifiaient selon l'activité partagée. Cette

théorie d'apprentissage local, illustrée par les écrits de **Donald Hebb** (1949), énonçait le principe suivant : lorsque deux neurones  $x_i$  et  $x_j$  s'activent simultanément, la liaison  $\omega_{i,j}$  se renforce. De façon mathématique, on peut schématiser ce phénomène par

$$\Delta\omega_{i,j} \propto x_i x_j,$$

où  $x_i$  et  $x_j$  désignent l'intensité d'activation de neurones  $i$  et  $j$ . Cette découverte suggérait déjà la possibilité d'une **auto-organisation** : un réseau de neurones où les pondérations se réajustent selon la co-occurrence, conduisant progressivement à la formation de motifs ou de regroupements stables.

La rencontre de la **cybernétique** et de la **neurosciences computationnelle** a ainsi ouvert la voie à des formalismes mathématiques permettant de décrire comment un **réseau** – qu'il s'agisse de neurones biologiques ou d'unités artificielles – peut se configurer de lui-même à travers des mises à jour strictement **locales**. Les principes de la rétroaction et de la plasticité synaptique convergeaient pour affirmer qu'un ordre global peut naître de simples règles d'ajustement, sans supervision. Cette conviction, déjà présente dans les prototypes informatiques modestes de l'époque, a servi de socle aux développements qui suivront : l'idée de "synergie" entre unités, chère au **Deep Synergy Learning**, hérite directement de la plasticité hébbienne et des mécanismes de contrôle cybernétique.

Bien que les moyens techniques fussent, à l'époque, limités, les écrits fondateurs évoquaient déjà la **stabilisation** d'un réseau autour de certaines combinaisons privilégiées d'activités. On y présentait la formation de **clusters** ou d'**assemblées** neuronales, précurseurs des théories d'**attracteurs** et de **cartes auto-organisées**. C'est dans ce terreau scientifique – porté par l'intérêt pour les systèmes **auto-régulés**, la découverte de la **plasticité** synaptique et le contexte de la **cybernétique** – qu'allait naître l'ambition de construire des réseaux véritablement **adaptatifs**, capables de s'**organiser** à partir de règles élémentaires. Le **Deep Synergy Learning**, bien plus tard, s'inscrira dans cette continuité : la mise à jour d'un poids  $\omega_{i,j}$ , la quête d'une **co-opération** locale et l'idée qu'un **ordre** global peut émerger de multiples interactions locales découlent étroitement de ces prémices.

#### 2.1.1.2. Naissance de l'Idée d'Organisation sans Superviseur dans Certains Laboratoires (années 1950–1960)

L'**apprentissage machine** des premières décennies, tel qu'illustré par le perceptron de Frank Rosenblatt (1958), se focalisait principalement sur des mécanismes **supervisés** reposant sur des *labels* explicites et un signal d'erreur orientant la mise à jour des poids. Cependant, dès la fin des années 1950, plusieurs laboratoires ont amorcé des recherches explorant la possibilité d'**organiser** les **réseaux** ou les **unités neuronales sans** recours à la supervision directe. L'intuition sous-jacente à ces travaux, influencée par la **cybernétique** et la biologie, était de laisser les entités neuronales, désignées par  $x_i$ , s'**auto-régler** par des **règles locales**, en se fondant sur la redondance ou la régularité intrinsèques des **données** d'entrée.

### A. Expériences Préliminaires issues de la Cybernétique

À la suite des réflexions initiées par Norbert Wiener, certains chercheurs se sont inspirés du principe de **rétroaction** pour imaginer des systèmes neuronaux capables de se **corriger** et de **s'ajuster** en l'absence d'un enseignant. Les premiers prototypes, bien que rudimentaires, comportaient un **petit réseau** de neurones soumis à des flux sensoriels. Les **pondérations** internes évoluaient progressivement en suivant des règles élémentaires de renforcement : si deux unités  $\mathbf{x}_i$  et  $\mathbf{x}_j$  coïncidaient fréquemment dans leur activation, le lien  $\omega_{i,j}$  se trouvait augmenté, formalisant ainsi un renforcement local. Ce mécanisme rappelle, dans une version simplifiée, l'idée d'une **co-occurrence** statistique où, mathématiquement, on peut écrire

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta \kappa(\mathbf{x}_i(t), \mathbf{x}_j(t)),$$

avec  $\eta$  un taux d'apprentissage local et  $\kappa(\cdot, \cdot)$  une fonction soulignant la co-activation simultanée des neurones  $\mathbf{x}_i$  et  $\mathbf{x}_j$ .

### B. Émergence du Concept d'Auto-Clusterisation

De premiers rapports internes (dans des laboratoires tels que le MIT ou l'Université Stanford) évoquaient la possibilité de **regrouper** automatiquement les données ou les neurones, même si le terme “clustering” n'était pas toujours formalisé de la sorte. Le concept de “résonance” ou de “coopération neuronale” commençait à circuler, pour désigner la faculté de certains sous-ensembles d'unités à présenter un comportement **synchronisé**. Cette synchronie, renforcée par un schéma de mise à jour local, tendait à **former** ce qu'on nommerait plus tard un **cluster**. On pressentait déjà qu'en exploitant uniquement la distribution des données — c'est-à-dire la *fréquence* et la *corrélation* des activations neuronales —, il était possible de dégager une structure latente sans qu'un label de sortie intervienne.

### C. Rupture par rapport aux Méthodes Supervisées

À la différence du perceptron de Rosenblatt, qui nécessitait un **signal d'erreur** pour orienter la correction des poids, ces laboratoires s'intéressaient à des systèmes pouvant **auto-exploiter** la simple densité ou co-activation des données d'entrée. D'un point de vue algorithmique, on sortait donc du paradigme

$$w_{\text{nouveau}} = w_{\text{ancien}} - \eta \frac{\partial E}{\partial w},$$

car la fonction d'erreur  $E$  était absente ou du moins implicite ; seule la **co-occurrence** neuronale venait guider l'ajustement. Ces expériences soulignaient le potentiel d'une **catégorisation spontanée**, c'est-à-dire la découverte de regroupements (catégories) sans recourir à un enseignant.

### D. Exemples de Règles Locales de Mise à Jour

Les discussions tournaient souvent autour du fait que la **corrélation** d'activation entre deux neurones  $\mathbf{x}_i$  et  $\mathbf{x}_j$  pouvait être utilisée comme critère de renforcement. Ainsi, la “loi de Hebb”, régulièrement citée dans ces travaux, posait qu'une activation simultanée doit se traduire par un **renforcement** de la connexion. Certains laboratoires introduisaient aussi une **inhibition latérale**, destinée à redistribuer les ressources et à éviter qu'un seul sous-réseau ne monopolise

l'apprentissage. Formulellement, on pouvait ajouter un **terme d'inhibition** proportionnel à la somme des activations concurrentes, conduisant à un rééquilibrage permanent :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta \left[ \mathbf{x}_i(t) \mathbf{x}_j(t) - \beta \sum_k \mathbf{x}_i(t) \mathbf{x}_k(t) \right],$$

où  $\beta$  est un paramètre d'inhibition, et la partie  $\sum_k \mathbf{x}_k(t)$  reflète la compétition entre neurones rivaux.

### *E. Bases d'une Future Formalisation*

Ces travaux pionniers n'ont pas immédiatement conduit à une théorie unifiée, en raison du manque de ressources computationnelles et de la jeunesse du formalisme mathématique. Toutefois, ils ont jeté les bases d'une **vision** selon laquelle la **donnée** en elle-même, pour peu qu'elle soit soumise à un réseau présentant un mécanisme de renforcement local, peut induire une **organisation spontanée**. L'histoire de l'IA retiendra que la fin des années 1950 et le début des années 1960 ont vu la **genèse** des prémices de l'**apprentissage non supervisé** et de l'**auto-organisation**, lesquelles s'affirmeront au cours des décennies suivantes dans des réalisations plus formelles, telles que les **cartes de Kohonen (SOM)** ou d'autres algorithmes de clustering non supervisé. Les fondements de cette dynamique trouvent aujourd'hui leur prolongement dans des approches modernes comme le **Deep Synergy Learning**, où les principes de **cohésion** et de **coopération** locale se traduisent par des **règles de mise à jour** autonomes et contextuelles, assurant la formation de sous-structures pertinentes sans l'intervention d'un enseignant externe.

#### *2.1.2. SOM, Hopfield et la Génération des Réseaux Associatifs*

Alors que la **cybernétique** et les premiers travaux d'auto-organisation “sans superviseur” (2.1.1) ont semé les graines d'une plasticité neuronale locale, la recherche sur les **réseaux associatifs** et les **Self-Organizing Maps (SOM)** dans les années 1970-1980 a considérablement renforcé l'idée qu'un système pouvait **découvrir** des structures en l'absence de labels explicites. Les modèles de Teuvo Kohonen (SOM) et de John Hopfield (réseaux associatifs) illustrent deux facettes majeures de l'auto-organisation : la **formation spontanée de cartes topologiques** et l'**association mémorielle** à travers des attracteurs stables. On verra dans ce sous-chapitre (2.1.2) comment ces approches, tout en se développant indépendamment, ont préparé le terrain pour des paradigmes plus récents tels que le **Deep Synergy Learning (DSL)**.

##### *2.1.2.1. Teuvo Kohonen : Self-Organizing Maps (SOM) et Topologie Émergente*

Les **Self-Organizing Maps (SOM)**, introduites par Teuvo Kohonen au début des années 1980, ont exercé une influence considérable dans l'**apprentissage non supervisé**. Leur principe repose sur la capacité d'un **réseau** à *projeter* un espace d'entrées (souvent de grande dimension) sur une **grille bidimensionnelle**, tout en conservant autant que possible les **relations de proximité** entre les données. Cette méthode constitue un jalon historique de l'**auto-organisation**, démontrant de manière tangible qu'un réseau neuronal peut découvrir des **structures** dans des données sans requérir de labels explicites.

### A. Préservation de la Topologie

L'idée fondamentale est de préserver la **topologie** des entrées dans l'espace de sortie. Les unités ou "neurones" de la carte sont disposés sous forme de grille (par exemple, deux dimensions), chaque neurone  $\mathbf{n}_i$  possédant un vecteur de poids  $\mathbf{w}_i$ . À chaque présentation d'un vecteur d'entrée  $\mathbf{x} \in \mathbb{R}^d$ , le neurone **gagnant**, souvent noté  $\mathbf{n}_{\text{BMU}}$ , est celui pour lequel la distance

$$\|\mathbf{x} - \mathbf{w}_{\text{BMU}}\|$$

est minimale. Les neurones voisins, définis par un **voisinage** décroissant au fil des itérations, sont **ajustés** pour se rapprocher de  $\mathbf{x}$ . Cette mise à jour peut s'exprimer, pour un neurone  $\mathbf{n}_j$  situé à proximité du gagnant  $\mathbf{n}_{\text{BMU}}$ , sous la forme :

$$\mathbf{w}_j(t+1) = \mathbf{w}_j(t) + \alpha(t) \mathcal{H}_{\text{BMU},j}(t) [\mathbf{x}(t) - \mathbf{w}_j(t)],$$

où  $\alpha(t)$  est un taux d'apprentissage, et  $\mathcal{H}_{\text{BMU},j}(t)$  un **facteur de voisinage**, généralement une fonction gaussienne de la distance entre  $\mathbf{n}_j$  et  $\mathbf{n}_{\text{BMU}}$  dans la grille. Par ce mécanisme, la carte se "**déforme**" localement pour accueillir l'information, créant ainsi une **continuité** qui reflète la structure sous-jacente des données.

### B. Apprentissage Auto-Organisé sans Labels

Le procédé s'inscrit clairement dans un **cadre non supervisé**. En l'absence de labels, la SOM repère des regroupements dans l'espace d'entrée tout en imposant une **cohérence topologique** sur la grille de sortie. De sorte qu'au fur et à mesure des itérations, le réseau s'organise de lui-même en attribuant à chaque neurone  $\mathbf{n}_j$  une zone de compétence. Les régions de la SOM finissent par représenter des **catégories** ou des **familles** de vecteurs. Il s'agit là d'un exemple marquant de la capacité d'un réseau à apprendre la structure latente des données sans information externe. Dans l'histoire de l'auto-organisation, ces travaux prolongent la philosophie décrite en section 2.1.1 au sujet des premiers prototypes sans superviseur.

### C. Impact sur la Vision de l'Auto-Organisation

Les cartes de Kohonen ont fait la démonstration qu'une **cartographie continue** des données pouvait émerger, d'une façon parfois rapprochée de la **corticotopie** observée chez l'animal ou l'humain (aires sensorielles). Des laboratoires contemporains ont développé en parallèle d'autres approches, comme les réseaux à attracteurs (Hopfield) ou les modèles neuronaux de Shun-Ichi Amari, élargissant encore la diversité des méthodes. L'importance historique des SOM tient à leur présentation didactique et à leur succès pour illustrer le **concept** de structure cognitive émergente. Elles ont ainsi balisé le chemin pour des algorithmes modernes d'**auto-organisation** dans les réseaux neuronaux.

### D. Liens Conceptuels avec le DSL

Le **Deep Synergy Learning (DSL)**, décrivant lui aussi un schéma d'ajustement auto-organisé des **liaisons** entre entités d'information, partage certains points communs avec la SOM. Dans le cas d'un SOM, la notion de **voisinage** sert de force de coordination locale. De façon similaire, au sein du DSL, la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  joue le rôle de régulateur pour décider du renforcement ou de l'affaiblissement d'un lien  $\omega_{i,j}$ . Les SOM peuvent être considérées comme un modèle relativement

**statique**, puisqu'elles imposent une grille préfixée et un voisinage décroissant dans le temps. Le DSL, en revanche, s'applique de manière plus **généraliste**, en laissant le réseau se **reconfigurer** en continu via des règles de mise à jour basées sur la synergie, ce qui autorise une plasticité plus large, y compris dans la **topologie** même des liens. Cependant, l'esprit est identique : aucune **supervision** n'est requise pour faire émerger un **ordre** dans la distribution des entités.

### *E. Vers des Extensions Multimodales et la Synergie*

En s'appuyant sur la faculté des SOM à regrouper des signaux divers, certains chercheurs ont considéré des **applications multimodales** (ex. signaux audio et images) dans lesquelles une carte conservait une **cohérence** entre différentes modalités. Bien que ces applications soient restées assez limitées en pratique, elles ont jeté les bases d'une réflexion sur la fusion de données hétérogènes et sur la possibilité qu'un réseau puisse **apprendre** à associer plusieurs sources. Cette logique se retrouve dans le DSL, qui pousse plus loin l'idée de **fusion coopérative** : en permettant à n'importe quelles entités de développer des liens flexibles selon leur **score de synergie**, il devient envisageable d'intégrer de multiples modalités de façon plus souple qu'une grille 2D imposée.

### *Conclusion*

Les **Self-Organizing Maps** de Kohonen représentent un tournant déterminant dans la formalisation de l'**auto-organisation**. Elles ont démontré la possibilité, pour un réseau, de cartographier sans labels explicites un vaste espace de données en une structure de sortie exploitable. L'importance pédagogique et les succès pratiques de la SOM ont entretenu le goût pour les systèmes **non supervisés** dans la communauté scientifique. Les idées directrices — réglage local, émergence de clusters, topologie imposée ou auto-imposée — ont ensuite nourri d'autres paradigmes à vocation plus générale, en particulier le **Deep Synergy Learning**, qui étend la notion de proximité et l'adapte à la formation continue de réseaux libres de leurs connexions et de leur organisation. Le SOM fut ainsi l'une des premières démonstrations solides de la **capacité** d'un réseau à **auto-structurer** la représentation de son environnement.

#### **2.1.2.2. John Hopfield : Mémoires Associatives, Attracteurs et Minima d'Énergie**

Les travaux de **John Hopfield**, menés au début des années 1980, ont introduit un modèle neuronal emblématique désigné sous le nom de **réseau associatif** ou "Hopfield network". Cette proposition, qui s'inspire d'idées analogues à celles de la **physique statistique** (systèmes de spins, modèles d'Ising ou de Potts), a mis en lumière la possibilité qu'un **réseau** formé d'unités simples puisse se stabiliser dans des **configurations** représentant des "mémoires" ou des **attracteurs**. La théorie démontre que, grâce à une série de **règles locales** d'activation et de mise à jour des poids, le réseau finit par adopter spontanément un **état** ordonné à partir d'une situation initiale partielle ou bruitée.

Sur le plan conceptuel, un réseau Hopfield se compose de neurones ou d'unités binaires, interconnectés de façon quasi complète, avec des poids  $\omega_{i,j}$ . Les poids sont déterminés de manière à ce que certains vecteurs cibles  $\mathbf{x}^*$  deviennent des **états stables** : si le système est initialisé dans une configuration voisine de  $\mathbf{x}^*$ , alors la dynamique conduira à une convergence vers ce même vecteur stable. Cette capacité illustre la notion de **mémoire associative**, où le réseau est en mesure de "retrouver" un motif mémorisé à partir d'informations incomplètes. La convergence résulte de la descente vers un **minimum d'énergie**, ce qui se formalise par une fonction  $E(\mathbf{x})$  dite "énergie



de Hopfield”, dont la décroissance au fil des itérations atteste de la stabilisation du réseau. On peut l’écrire, par exemple, comme

$$E(\mathbf{x}) = -\frac{1}{2} \sum_{i \neq j} \omega_{i,j} x_i x_j$$

Lorsque  $x_i \in \{-1, +1\}$  sont les états des neurones.

L’idée d’associer une “**énergie**” à chaque configuration, empruntée aux modèles de spins en physique, a contribué à la compréhension de la **dynamique neuronale** : l’organisation globale émerge de l’itération de règles locales, chaque unité cherchant à minimiser le niveau d’énergie via l’actualisation de son état. Les mémoires stockées correspondent à des **minima locaux** de l’énergie, appelés **attracteurs**, vers lesquels converge le système si l’état initial se trouve dans leur bassin d’attraction. Cette observation a conforté le principe selon lequel un ordre collectif peut naître en l’absence d’un **superviseur** extérieur, du fait même de l’interaction **coopérative** entre unités.

D’un point de vue **auto-organisation**, le réseau de Hopfield n’exige pas de labels explicites pour récupérer un motif partiellement effacé ou bruité. Les poids peuvent être calculés par des règles associatives simples (souvent de type Hebb), mais une fois installés, la dynamique interne se déroule sans aide extérieure. Cette démarche illustre la logique d’**émergence** : la configuration finale du réseau, perçue comme la “mémoire récupérée”, ne se laisse pas dicter par une consigne imposée à chaque étape, mais relève plutôt d’une évolution **naturelle** vers un état stable. On rejoint ici les idées qui seront approfondies dans des approches contemporaines d’**apprentissage non supervisé** ou d’**auto-organisation** (voir section 2.1.2.1 pour la SOM de Kohonen).

Les liens avec le **Deep Synergy Learning (DSL)** peuvent être soulignés. Dans un réseau Hopfield, on observe une **coopération** entre unités guidée par l’énergie globale, et la convergence dans un état stable évoque la stabilisation d’un cluster de neurones. De façon analogue, dans le DSL, la présence d’une **fonction de synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  joue un rôle comparable : lorsque deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  partagent une interaction jugée élevée, leurs liens se renforcent et le réseau évolue vers des groupements cohérents. Les mécanismes d’ajustement des pondérations, dans le DSL comme dans le modèle de Hopfield, traduisent une dynamique **auto-organisée** conduisant à l’apparition d’**attracteurs collectifs** ou de **sous-structures** stables.

La perspective de **minima d’énergie** s’applique dans le DSL lorsqu’on étudie la convergence des poids  $\omega_{i,j}$  vers des configurations privilégiées. Bien que la formulation soit plus générale dans le cas du DSL (synergies n-aires, multimodalité, structure libre du réseau), le point essentiel demeure : un ensemble de règles locales peut suffire à faire émerger un **ordre global**, sans supervision externe. Le modèle de Hopfield constitue donc un **antécédent** conceptuel crucial, montrant comment des processus neuronaux simplifiés, hébergeant une logique énergétique, peuvent conduire à l’apparition de représentations stables. Cet héritage se retrouve dans la conception du DSL, où la “synergie” prolonge la notion d’affinité ou de corrélation, et où la plasticité multientités s’interprète comme une “generalisation” de la mémoire associative à des contextes et des modalités plus complexes.

### 2.1.2.3. Comparaison Partielle avec la Logique “Hebbienne”

Les **Self-Organizing Maps** de Kohonen et les **réseaux associatifs** de Hopfield, analysés précédemment dans les sections 2.1.2.1 et 2.1.2.2, illustrent deux approches phares de l'**apprentissage non supervisé** centrées sur la découverte de *patterns* ou de configurations stables en l'absence de labels explicites. Ces travaux s'inscrivent dans une continuité historique marquée par la **logique “Hebbienne”**, introduite dès la fin des années 1940. Le principe énoncé par Donald Hebb évoque que deux neurones qui s'activent conjointement renforcent leur connexion. La présente section (2.1.2.3) discute les liens qui unissent ces trois orientations (SOM, Hopfield, Hebb) et montre en quoi elles partagent des fondements communs tout en se distinguant dans leur implémentation et leur formalisme.

#### A. Fondements de la Règle de Hebb

Au cœur de la théorie de Donald Hebb, exposée dans *The Organization of Behavior* (1949), se trouve l'idée qu'un lien entre deux neurones  $\mathbf{x}_i$  et  $\mathbf{x}_j$  se renforce dès lors que leurs activations se produisent en simultané. L'intuition peut se formaliser par une **règle de mise à jour** locale de la forme

$$\Delta\omega_{i,j} \propto \mathbf{x}_i \mathbf{x}_j,$$

où  $\omega_{i,j}$  désigne la connexion synaptique entre les neurones  $i$  et  $j$ . L'esprit est qu'une activation conjointe déclenche un **renforcement**, alors qu'une activation dissociée n'engendre pas d'augmentation (voire induit une réduction) de cette liaison.

#### B. Similarités avec les SOM et les Réseaux Associatifs

Les **Self-Organizing Maps** (SOM) recourent à un mécanisme de renforcement lorsqu'un neurone “gagne” face à un vecteur d'entrée. Il y a un rapprochement entre le neurone vainqueur  $\mathbf{n}_{\text{BMU}}$  et le vecteur d'entrée, ainsi qu'un ajustement pour les neurones voisins dans la grille. Cette mise à jour rappelle l'idée de co-activation : le neurone et la donnée se retrouvent “associés”, ce qui ressemble, dans un cadre discret, à la logique “hebbienne” de corrélation locale. De même, les **réseaux associatifs** de Hopfield adoptent une conception selon laquelle les poids  $\omega_{i,j}$  reflètent la *corrélation* ou la *co-occurrence* entre les unités  $\mathbf{x}_i$  et  $\mathbf{x}_j$ . Lorsqu'on “enregistre” un motif, la force des liaisons associées augmente selon un principe analogue à la loi de Hebb, rendant possible la récupération dudit motif lorsque le réseau est présenté à un fragment ou à un état bruité proche.

#### C. Divergences et Élargissements

Ces trois approches conservent toutefois des particularités marquées. Les SOM, par exemple, imposent une **dimension topologique** à travers la grille 2D et la notion de “voisinage”, un élément que la loi de Hebb seule n'intègre pas de manière explicite. Dans un **réseau Hopfield**, la logique d'**énergie** et de **minima** assure la stabilité globale, tandis que la règle hebbienne se décline plutôt comme un simple *update local*  $\Delta\omega_{i,j} \propto \mathbf{x}_i \mathbf{x}_j$ . L'idée d'une fonction d'énergie globale, introduite par Hopfield, généralise l'intuition de la co-occurrence vers une perspective plus large d'attracteurs. Enfin, il existe une multitude de variantes “hebbiennes” qui ajoutent des mécanismes de normalisation, de saturation ou d'inhibition, permettant un **auto-équilibrage** ou une répartition des ressources. Les SOM et les réseaux Hopfield constituent chacun une **implémentation** plus

spécifiquement contrainte : préservation topologique pour les SOM, attracteurs d'énergie pour Hopfield.

#### *D. Portée pour le Deep Synergy Learning*

Le **Deep Synergy Learning (DSL)** perpétue l'héritage "hebbien" en privilégiant un **score de synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  qui détermine si deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  renforcent ou affaiblissent leur lien  $\omega_{i,j}$ . Lorsque la synergie est élevée, on obtient une croissance de  $\omega_{i,j}$  proche de

$$\Delta\omega_{i,j} \sim \eta[S(\mathcal{E}_i, \mathcal{E}_j)].$$

Cette démarche rappelle étroitement l'esprit de la règle de Hebb, tout en permettant des **généralisations** (par exemple, co-information n-aire, interactions complexes) qui dépassent la simple multiplication  $\mathbf{x}_i \mathbf{x}_j$ . Ainsi, le DSL embrasse la dynamique auto-organisée du type "hebbien" mais l'adapte à une variété de contextes, notamment multimodaux, et autorise la reconfiguration de la **structure du réseau**.

#### *E. Conclusion sur la Continuité Historique*

Le rapprochement entre les SOM, les réseaux Hopfield et la règle de Hebb met en évidence un **tronc commun** : la conviction que l'**auto-organisation** peut être guidée par la co-activation ou la co-occurrence d'entités, sans étiquettes supervisées. Les SOM instaurent une topologie manifeste, Hopfield introduit le formalisme de l'énergie, tandis que la loi de Hebb s'énonce comme un canevas local et minimaliste. Le **Deep Synergy Learning**, en s'inscrivant dans cette tradition, étend la logique hebbienne à des **fonctions de synergie** plus sophistiquées, conférant au processus d'apprentissage une plasticité encore plus large et un champ d'application élargi. De même, il hérite la philosophie d'**auto-régulation** non supervisée, où la **coordination globale** émerge de la simple itération de règles d'interaction locale. Les paradigmes SOM et Hopfield ont donc grandement contribué à forger un terreau conceptuel que le DSL exploite pour proposer une vision plus englobante de l'**auto-organisation** sans supervision explicite.

#### *2.1.3. Entre Neurosciences Computationnelles et Physique Statistique*

Après avoir examiné les apports de l'approche "sans superviseur" (2.1.1) et les modèles de réseaux associatifs ou de cartes topologiques (2.1.2), on constate que l'**auto-organisation** a également bénéficié d'un important soutien théorique et conceptuel depuis la **neurosciences computationnelles** et la **physique statistique**. Ces deux champs, bien que distincts dans leurs méthodes, ont convergé dans l'idée que des interactions **locales** entre unités (neurones, spins, etc.) peuvent générer des **motifs** ou des **structures** émergentes. Ce sous-chapitre (2.1.3) discute des influences conjointes de la dynamique neuronale (2.1.3.1) et des modèles de la physique hors équilibre, formant ainsi un socle pour de futures approches telles que le **Deep Synergy Learning (DSL)**.

##### *2.1.3.1. Contribution des Modèles Biologiques (Dynamique des Neurones, Synapses)*

Les avancées en **neurosciences computationnelles**, dès les années 1960–1970, ont exercé une influence considérable sur la compréhension et la formalisation de l'**auto-organisation** en intelligence artificielle. Au-delà de la seule logique "hebbienne", ces travaux se sont inspirés de

mécanismes plus riches, tels que l'**inhibition latérale**, l'**homéostasie** ou encore les **modulations neuromodulatrices**. Les recherches d'Amari, de Grossberg et d'autres pionniers ont éclairé la manière dont la dynamique neuronale, régie par un ensemble d'**équations différentielles** ou de **règles itératives**, peut engendrer une **organisation spontanée** d'assemblées de neurones dans le cortex.

Un premier axe s'est centré sur la description mathématique de l'**activation neuronale**, souvent abordée via des champs neuronaux. Dans cette optique, on considère qu'un neurone  $x_i$  suit une équation de la forme

$$\frac{d x_i}{d t} = -\alpha x_i + \sum_{j \neq i} \left( w_{ij} f(x_j) \right) - \gamma g(x_i) + \theta_i,$$

où  $\alpha$  est un terme de décroissance,  $\sum_{j \neq i} w_{ij} f(x_j)$  représente la somme des influences (excitatrices ou inhibitrices) venant d'autres neurones,  $\gamma g(x_i)$  un terme d'homéostasie ou de saturation globale, et  $\theta_i$  un éventuel seuil ou un apport externe. Les valeurs  $w_{ij}$  constituent les **connexions synaptiques** entre neurones  $i$  et  $j$ . L'ensemble de ces équations, pris simultanément pour tous les neurones, fait apparaître des comportements comme l'**inhibition latérale** ou la **stabilisation** de patterns d'activité, propriétés fondamentales d'une auto-organisation neuronale.

Un second axe a porté sur la formation de **micro-assemblées neuronales** capables de se synchroniser et de se stabiliser de façon autonome, en particulier lorsque plusieurs neurones présentent une **co-activation** récurrente. Les neuroscientifiques ont documenté l'existence de tels groupes dans le cortex, où ils semblent représenter des stimuli spécifiques ou des fragments de concepts. Sur le plan algorithmique, on comprend que de simples règles de mise à jour locale des poids peuvent conduire à l'émergence de clusters : si deux neurones  $x_i$  et  $x_j$  s'activent souvent ensemble, alors la synapse  $\omega_{ij}$  se renforce, approximant la célèbre équation de type

$$\Delta \omega_{i,j} \propto x_i x_j.$$

Les neurones qui s'alignent ainsi sur la même fréquence de décharge forment une entité collective, ou **assemblée**, reflétant l'existence d'un pattern stable. C'est un mécanisme incontournable de la plasticité neuronale qui anticipe l'essentiel des principes d'**apprentissage non supervisé**.

Les modèles biologiques ont en outre démontré la nécessité de gérer la plasticité à plusieurs niveaux : la simple corrélation positive entre neurones, prolongement direct de la règle de Hebb, doit être modulée par des influences globales, par exemple l'**inhibition** généralisée (pour empêcher un groupe de neurones de devenir surdominant), ou l'**homéostasie** garantissant que l'activité moyenne du réseau se maintient dans une plage stable. Sur le plan formel, on peut introduire une équation supplémentaire qui régit l'évolution d'un paramètre global  $\mu$  (un niveau d'homéostasie), que l'on couple à la dynamique neuronale :

$$\frac{d \mu}{d t} = \kappa(\bar{x} - \mu),$$

où  $\bar{x}$  représente l'activité moyenne du réseau. De telles considérations ont enrichi la formulation de la **plasticité synaptique**, en montrant qu'elle n'est pas uniquement dictée par la corrélation entre neurones, mais qu'elle peut être encadrée par des mécanismes correcteurs à grande échelle.

Cette perspective multi-niveaux et la logique de **rétroaction** entre neurones ont profondément marqué le développement de l'**auto-organisation** en IA. Les travaux décrits dans la section précédente (SOM, réseaux Hopfield, etc.) trouvent en effet leur fondement dans l'idée qu'un **ordre** global peut émerger d'interactions locales, inspirées des phénomènes biologiques. Les paradigmes d'apprentissage non supervisé, y compris le **Deep Synergy Learning (DSL)**, en hériteront largement. Dans ce dernier, la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  généralise la corrélation neuronale à des entités plus variées que des neurones biologiques, tout en gardant l'esprit de **cohésion** et de **co-activation**. On y retrouve, de surcroît, la possibilité de gérer des influences extérieures ou des régulations globales sur la distribution des pondérations  $\omega_{i,j}$ , s'apparentant à ces mécanismes d'homéostasie étudiés en neurobiologie.

Ces contributions biologiques éclairent la **logique** des règles de mise à jour et les **régularisations** nécessaires pour rendre un réseau de neurones, ou plus généralement un **réseau auto-organisé**, à la fois flexible et stable. L'inhibition latérale, la compétition, la notion d'assemblées synchronisées, et la multi-échelle de plasticité guident les solutions mises en œuvre dans de nombreux algorithmes d'auto-organisation. Comme expliqué en 2.1.3.2, la **physique statistique** apportera un complément de formalisation, en introduisant le concept de fonction d'énergie globale et de transitions de phase, pour appuyer cette thèse selon laquelle de simples lois locales peuvent faire naître une structure cohérente.

### 2.1.3.2. Apports de la “Synergetics” (Haken) et de la Thermodynamique Hors Équilibre

Le rôle de la **physique** dans la formalisation de l'**auto-organisation** s'est avéré tout aussi déterminant que celui des **neurosciences computationnelles** (cf. section 2.1.3.1). Les réflexions autour de la “**synergetics**”, initiées par Hermann Haken, et les avancées de la **thermodynamique hors équilibre**, mises en lumière notamment par Prigogine, ont considérablement enrichi la compréhension de la **structure émergente** dans des systèmes complexes. Ces idées, apparues dans les années 1970, ont permis de décrire la formation **spontanée** de motifs ou de clusters stables, en expliquant comment des interactions locales peuvent, par amplification mutuelle, donner naissance à un *ordre global*. Le **Deep Synergy Learning (DSL)** tire un profit conceptuel de ces travaux, en prolongeant le cadre d'analyse proposé par la synergetics pour rendre compte des dynamiques de pondérations et de synergies entre entités hétérogènes.

Un premier apport essentiel de la synergetics réside dans l'introduction de la notion de **paramètre d'ordre**, qui désigne une grandeur globale traduisant le passage d'un état désordonné à un état ordonné. Dans un système physique, ce paramètre se manifeste, par exemple, dans l'intensité du rayonnement laser ou la convection de Rayleigh-Bénard, où une **bifurcation** engendre l'émergence d'un motif spatial cohérent. D'un point de vue mathématique, l'auto-organisation se formalise ainsi :

$$\frac{dx_i}{dt} = F(x_i, \{x_j\}_{j \neq i}, \lambda),$$

où  $x_i$  représente l'état d'une unité  $i$  et  $\lambda$  est un **paramètre de contrôle**. Au-delà d'un certain  $\lambda_{\text{critique}}$ , une fluctuation initiale peut se retrouver amplifiée jusqu'à donner un **état global** ordonné, indiqué par l'évolution d'un paramètre d'ordre. Dans une perspective d'**apprentissage**

**automatique**, cette situation correspond à la formation d'un **cluster** ou d'une **représentation collective**, résultant de boucles de rétroaction locales entre unités.

La **thermodynamique hors équilibre** étudie, quant à elle, les systèmes où un **flux** permanent d'énergie ou de matière empêche la stabilisation à un équilibre classique. Les travaux de Prigogine ont, par exemple, mis en avant la formation de **structures dissipatives**, capables de s'auto-organiser grâce à l'échange d'énergie avec l'environnement, comme le suggère l'apparition de vortex dans un fluide ou d'oscillations chimiques Belousov-Zhabotinsky. Le parallèle avec un **réseau** d'unités neuronales ou d'agents est naturel : si chaque entité subit un influx permanent de données ou de signaux, l'apparition de **clusters** cohérents peut être lue comme une transition hors équilibre, portée par une fonction d'**énergie** ou de **quasi-énergie**. Dans une telle vision, l'évolution d'un réseau adaptatif se conçoit à travers l'équation

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(\mathcal{E}_i, \mathcal{E}_j) - \tau \omega_{i,j}(t)],$$

où la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  prend la place d'un "moteur" non linéaire, apte à déclencher une **bifurcation** quand certaines liaisons excèdent un "seuil critique".

Cette analogie se trouve renforcée par l'**interprétation énergétique**. Dans la synergetics, on peut décrire un système par une fonction de potentiel (ou d'énergie libre) qui décroît au fur et à mesure qu'un ordre s'établit. De la même manière, le **DSL** assimile la dynamique des poids  $\omega_{i,j}$  à la recherche d'un certain **ordre** (ex. configuration de clusters) reflétant un minimum de quasi-énergie ou un optimum local en matière de synergie. Le **bruit**, au sens physique, peut alors jouer un rôle crucial : un léger déséquilibre peut suffire à amorcer la croissance d'un cluster, et un niveau modéré de perturbation autorise le réseau à s'extraire de minima sous-optimaux.

Les **applications** de cette métaphore sont variées. Dans le **DSL**, dès lors qu'on interprète les entités comme des "particules informationnelles" interagissant via  $S(i, j)$ , la transition d'un état dispersé à un état **agrégé** (clusterisé) s'apparente à un passage de phase. Les liens  $\omega_{i,j}$  s'accroissent jusqu'à constituer un sous-réseau stable, tandis que les connexions marginales disparaissent. Cette émergence graduelle peut faire l'objet d'une analyse semblable aux **transitions de phase** en physique, soulignant l'universalité d'un mécanisme auto-organisé où un ordre global naît d'un **feed-back** local.

Dans la conception qu'en donnent les promoteurs du DSL, on retrouve la volonté de **prolonger** l'héritage hakenien : rechercher un "**ordre stable**" (clusters ou attracteurs) à partir d'interactions **locales** successives, plutôt que d'imposer a priori une architecture rigide. Cette parenté s'illustre particulièrement bien lorsque la mise à jour des pondérations s'interprète comme une **minimisation** d'une forme d'énergie, associée à la diffusion des **synergies**  $S(\mathcal{E}_i, \mathcal{E}_j)$ . L'inspiration de la thermodynamique hors équilibre conduit, en outre, à admettre que de faibles perturbations ponctuelles peuvent déclencher de grandes réorganisations, gage d'une **plasticité** et d'une capacité à "trouver" l'ordre dans le désordre.

Ainsi, l'apport de la **physique statistique** et de la **synergetics** fournit à l'auto-organisation un **formalisme** qui accentue la notion de **transition** et de **paramètre d'ordre**. Les paradigmes ultérieurs, comme le **Deep Synergy Learning**, reprennent les principes de boucles de rétroaction positive et d'amplification des synergies, en remplaçant les interactions spin par des **synergies** plus complexes adaptées à divers types d'entités (ex. vecteurs, symboles, flux temporels). La logique



sous-jacente demeure toutefois la même : l'**ordre** émerge de multiples mises à jour locales, et le système peut franchir des paliers critiques pour se stabiliser en configurations cohérentes et **auto-organisées**.

### 2.1.3.3. Premiers Parallèles avec la Notion de Synergie dans un Réseau d'Entités

Les avancées exposées en **neurosciences computationnelles** (section 2.1.3.1) et en **physique statistique** (section 2.1.3.2) ont fait ressortir un principe commun, un **réseau** composé de multiples entités (neurones, spins, agents) peut s'auto-organiser via l'accumulation d'**interactions locales**. Cette double perspective – dynamique neuronale et thermodynamique hors équilibre – a conduit à employer le terme de “**synergie**” pour décrire la force ou l'intérêt mutuel reliant deux unités dans le réseau. Les points suivants (2.1.3.3.1 à 2.1.3.3.4) soulignent comment, dès ces premiers travaux, on peut rapprocher cette démarche de ce qui sera plus tard le **Deep Synergy Learning (DSL)**.

#### *A. Passer du Vocabulaire de la Physique au Vocabulaire Neuronal*

Dans les modèles Hopfield ou Ising, on définit une **énergie** globale mesurant la cohérence ou, à l'inverse, le désordre. Les entités y sont liées par des **couplages**  $\omega_{ij}$ . De la même manière, la **synergetics** (Haken) postule qu'un **paramètre d'ordre** peut émerger lorsque les connexions se renforcent collectivement. On a alors suggéré de passer d'une lecture “énergie négative” ou “coupling” à la notion plus “positive” de **synergie**, où celle-ci exprimerait la “**plus-value**” constructive de la relation entre deux unités. En neurosciences, il est fréquemment question de “**gain**” ou d’“affinité” entre neurones, soulignant la co-activation productive. Dans la lignée de la plasticité synaptique, la co-occurrence neuronale a souvent été décrite comme un facteur de renforcement local, certains articles utilisant explicitement la terminologie “**co-information**” ou “**co-opération**”. Ainsi, les premières bases d'une “synergie” ont été posées par analogie avec la description de l'énergie dans les systèmes physiques.

#### *B. Hypothèses de Lien Local-Global*

Les théoriciens de la **synergetics** ont mis l'accent sur le fait que, si la “force” locale entre deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  est suffisamment élevée, elles tendent à se synchroniser ou à coopérer. Au fur et à mesure que ce phénomène se propage dans l'ensemble du réseau, on assiste à la formation d'un **état ordonné** (un cluster, un attracteur, un domaine de spins, etc.). La **coopération** locale est donc censée produire un **ordre global** via une accumulation de petites interactions. Cette dynamique se retrouve tant dans la formation d'assemblées neuronales (en neurosciences) que dans la constitution de domaines magnétiques (en physique), ou encore dans le principe d'un **réseau** qui franchit un **seuil** pour consolider des groupes fortement connectés. La synergie, en se focalisant sur la “valeur constructive” de la liaison entre deux unités, sous-entend que l'effet global est un gain pour l'organisation du système. On peut alors parler d'une transition structurelle, où le **paramètre de contrôle** (température, taux  $\eta$ ) déclenche une réorganisation.

Certains modèles de champs neuronaux introduisent un poids  $\omega_{ij}(t)$  dépendant de la corrélation entre les signaux d'activation, ce qui s'apparente déjà à une **mesure de synergie**. Dans le domaine de la physique, des études soulignent qu'une “mise en phase” (phase-locking) entre deux oscillateurs peut être quantifiée par un indice de co-activation, lequel s'approche également de l'idée de synergie. Dans ce contexte, des heuristiques simplifiées s'énoncent sous forme de règles locales, si la corrélation entre deux unités dépasse un certain seuil, le lien se renforce ; dans le cas contraire, il s'affaiblit. Bien que rudimentaires, ces règles ouvrent la voie à un cadre plus général,

comme celui qu’adoptera le **DSL**. Une fonction  $S(i, j)$  exprimant le degré d’interaction “utile” ou “cohérente” entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . La démarche se concrétise en améliorant le simple critère “corrélation positive” vers d’autres formes de co-information.

L’idée de **réseaux dynamiques** où les liens  $\omega_{i,j}(t)$  évoluent en continu pour mettre en avant ou en retrait certaines connexions tire parti de cette notion naissante de synergie. Les modèles Hopfield, par exemple, conservent leurs poids constants après un certain apprentissage, alors que la **SOM** (Kohonen) s’appuie sur un voisinage prédéfini. Par contraste, dès qu’on conçoit la synergie comme une **variable** interne évoluant librement, le réseau s’apparente à un **système hautement plastique**. Il abandonne un cadre statique en faveur d’une mise à jour incessante, suivant

$$\omega_{i,j}(t + 1) = \omega_{i,j}(t) + \eta[S(i, j) - \tau \omega_{i,j}(t)].$$

Ce schéma préfigure le **Deep Synergy Learning**, où les entités ne sont plus restreintes aux neurones biologiques ou à des spins, mais peuvent inclure des capteurs, des modules symboliques ou d’autres agents. Le concept de **synergie** s’y généralise, passant d’une simple co-activation à une interaction multimodale ou multiniveau. De surcroît, la *structure du réseau* elle-même se trouve remodelée, la synergie permettant de dépasser le canevas fixe des SOM ou des réseaux associatifs classiques.

En définitive, la **convergence** entre neurosciences et physique a forgé la notion d’une synergie locale provoquant un ordre global. Les premières approches ont implémenté ce concept sous forme de règles de corrélation, de mesures de co-information ou de couplages spin, suscitant le projet d’un **réseau auto-organisé** dans lequel l’évolution des liaisons répondrait à un principe de “plus-value” coopérative. Le **DSL** émerge dans cette continuité, proposant un cadre unificateur où la synergie prend des formes variées et se règle en permanence, traduisant la capacité du réseau à **former, rompre, ou stabiliser** ses clusters d’entités en l’absence de supervision.

#### 2.1.4. Précurseurs de la Synergie Multimodale

Les recherches en **auto-organisation** se sont longtemps concentrées sur des données relativement homogènes (spectres sensoriels similaires ou vecteurs numériques). Cependant, dans de nombreux contextes biologiques et robotiques, des **modalités** distinctes (vision, toucher, audition, etc.) interagissent simultanément, incitant les chercheurs à imaginer des **systèmes** capables de fusionner plusieurs types de signaux au sein d’un même réseau adaptatif. Ce sous-chapitre (2.1.4) met en lumière comment, avant même l’essor du **Deep Synergy Learning (DSL)**, des équipes ont tenté de faire coexister différentes sources d’information dans des architectures orientées auto-organisation.

##### 2.1.4.1. Initiatives Cherchant à Fusionner Plusieurs Types de Données (Vision + Tactile, etc.)

Dès les premiers travaux en **robotique sensorielle**, l’idée d’une **multimodalité** s’est imposée comme un enjeu central. En effet, un robot ou un système cognitif peut être pourvu simultanément de caméras (capteurs visuels) et de senseurs tactiles (capteurs de pression, vibration), et souhaiter **découvrir** de manière autonome des corrélations ou des “**synergies**” entre ces différents flux. Le besoin de modéliser cette complémentarité, sans supervision explicite, a mené à diverses



expérimentations démontrant que la **co-activation** de canaux sensoriels peut se traduire par un **renforcement** des liens entre entités, selon un principe déjà évoqué en section 2.1.3.

D’anciens laboratoires ont notamment testé, dès la fin des années 1970, l’association de **réseaux neuronaux rudimentaires** à des données issues d’**images** et de **capteurs tactiles**, dans le but de reconnaître un objet ou de caractériser une scène. Le fonctionnement de ces prototypes consistait à considérer l’activation simultanée d’une zone visuelle  $\mathbf{v}_i$  et d’une zone tactile  $\mathbf{t}_j$ . Dans un schéma de type hebbien, la pondération  $\omega_{(v_i, t_j)}$  augmentait si ces deux entités coïncidaient lors de la perception d’un même objet. On peut formuler la mise à jour comme

$$\Delta\omega_{(v_i, t_j)} \propto \mathbf{v}_i \mathbf{t}_j,$$

indiquant que la corrélation entre la voie visuelle  $\mathbf{v}_i$  et la voie tactile  $\mathbf{t}_j$  renforce la liaison les unissant. À mesure que ces expériences étaient réitérées, la liaison correspondante se consolidait jusqu’à se traduire par une coopération plus robuste entre les deux modalités.

Dans le sillage des **Self-Organizing Maps (SOM)** proposées par Teuvo Kohonen, certains chercheurs ont implémenté des “SOM double”, consistant en deux grilles topologiques (l’une pour la vision, l’autre pour le tactile) reliées par un **mécanisme de pont**. Lorsque le même objet était perçu sur les deux modalités, le **voisinage** dans la grille visuelle et la grille tactile incitait chacune de ces deux cartes à se rapprocher, créant une **zone bimodale** où les neurones se trouvaient mutuellement renforcés. Bien que limités, ces essais de **fusion** sans label explicite démontraient déjà le concept qu’une **auto-organisation** conjointe peut conduire à des sous-zones où les informations visuelles et tactiles se *recouvrent*, facilitant une identification plus fiable des objets manipulés.

Parallèlement, les avancées en **neurosciences** révélaient l’existence d’aires associatives “**amodales**” dans le cortex, suggérant que la **coopération** des signaux visuels et tactiles peut survenir naturellement pour former une **perception** intégrée. Les modèles computationnels s’inspiraient de ce constat : à chaque co-occurrence de signaux, on augmentait la force d’un lien (ou d’un cluster) reliant ces deux canaux, ce qui, d’un point de vue mathématique, revient à généraliser la corrélation binaire vers une notion de co-activation dans le temps. De cette manière, l’apprentissage non supervisé exploitait la co-occurrence répétée des mêmes événements sur deux voies sensorielles, sans qu’il soit nécessaire qu’un humain étiquette explicitement leur correspondance.

Cette démarche de **renforcement** adaptatif entre flux hétérogènes préfigure la dynamique mise en œuvre dans le **Deep Synergy Learning (DSL)**. Au lieu de se restreindre à un couplage stationnaire (une SOM visuelle, une SOM tactile, etc.), on peut imaginer un **réseau** où les liens  $\omega_{(i,j)}(t)$  entre n’importe quelles entités — visuelles, tactiles, auditives ou symboliques — évoluent en continu, selon une fonction de synergie  $S(i,j)$  reflétant leur degré d’activation conjointe. La mise à jour suit un schéma tel que

$$\omega_{(i,j)}(t+1) = \omega_{(i,j)}(t) + \eta[S(\mathcal{E}_i, \mathcal{E}_j) - \tau \omega_{(i,j)}(t)],$$

ce qui rejoint la logique des heuristiques multimodales employées dans ces premiers prototypes. Ces recherches antérieures se révèlent donc être un **préalable direct** à la formulation du DSL, car elles ont posé les bases d’un **apprentissage réellement non supervisé**, tirant parti de la

complémentarité entre différents flux sensoriels ou conceptuels, et structurant spontanément un **réseau** où les pondérations fortes dénotent une intense **synergie** entre modalités.

En somme, avant l'avènement du **Deep Synergy Learning**, divers travaux en robotique et en neurosciences computationnelles avaient démontré la viabilité d'une auto-organisation **multimodale**, où le couplage entre modalité visuelle et tactile, notamment, se révélait un cas d'école. L'observation de corrélations temporelles ou fonctionnelles suffisait à susciter un **apprentissage** de type clustering, aboutissant à des représentations plus riches et plus stables que l'approche unimodale. Les idées ainsi expérimentées ont confirmé qu'il n'était pas nécessaire de disposer d'un **superviseur** externe pour associer deux flux distincts : la co-occurrence ou la convergence récurrente s'avérait largement suffisante pour laisser **émerger** la liaison adaptative souhaitée. Ce constat précède et anticipe directement les principes du DSL, qui pousse plus loin la logique de **synergie adaptative** dans un cadre unifié et potentiellement extensible à plusieurs entités ou modalités simultanées.

#### 2.1.4.2. Influences sur la Mise en Place d'un Cadre Unifié de Synergie Adaptative

Les expériences décrites en section 2.1.4.1, qu'il s'agisse de **fusion** vision+tactile ou de cartes "double-SOM" connectées, ont fait émerger une nécessité : disposer d'une **approche générale** pour décrire la manière dont plusieurs **modalités** (ou entités hétérogènes) peuvent coopérer par **auto-organisation**. Les initiatives multimodales pionnières ont souligné l'intérêt d'étendre les règles traditionnelles d'apprentissage sans superviseur vers des situations où les **entités** (capteurs, flux symboliques, représentations vectorielles) ne partagent pas forcément le même format ni le même espace de description. Les réflexions suscitées dans ce domaine ont contribué à l'élaboration ultérieure de paradigmes tels que le **Deep Synergy Learning (DSL)**, dont la logique unifie et **généralise** les notions d'interactions adaptatives entre entités variées.

##### *A. Impulsion depuis la Synergie "Bimodale"*

En mettant en relation, par exemple, des données **visuelles** (images) et **tactiles** (capteurs de pression), plusieurs laboratoires ont observé que les **règles classiques** (du type corrélation ou co-activation) se révélaient efficaces même lorsque les flux appartenaient à des espaces différents. Au lieu de se limiter à une **similarité** calculée dans un espace vectoriel identique, ces chercheurs ont proposé des fonctions de "**matching**" ou de "distance inter-modale", aptes à comparer un patch visuel à un signal tactile. On voyait ainsi qu'il suffisait d'avoir une mesure d'**affinité** cohérente pour permettre un renforcement local, selon un principe semblable à :

$$\Delta\omega_{i,j} \propto \text{similarité}(\mathbf{x}_i, \mathbf{x}_j).$$

Cette conclusion a incité la communauté à considérer des fonctions encore plus flexibles, où chaque couple d'entités définirait sa propre notion de corrélation ou de distance, préparant le chemin vers une **synergie** plus générale.

##### *B. Vers une Extension n-aire*

À partir de la fusion de deux modalités, un passage naturel a consisté à s'interroger sur la possibilité de connecter **plusieurs** flux simultanément. Dès lors, on envisageait une **synergie n-aire**, où au-delà d'un simple produit binaire (vision + tactile), on permettait la coopération de la vision, du toucher, de l'audition, voire d'autres sources (signaux proprioceptifs, informations symboliques, etc.). Cette extension laissait entrevoir des perspectives de **cohésion collective** : un groupe d'entités

pouvait se synchroniser autour d'un même événement, même sans label externe, par simple co-occurrence en temps ou en contexte. Les premiers essais, bien qu'expérimentaux, ont montré qu'un réseau pouvait découvrir des regroupements plus sophistiqués dès lors qu'il disposait de **flux variés** et d'une mise à jour adaptative des liens. Sans le cadre théorique unifié, la mise en œuvre restait toutefois ad hoc.

### C. Nécessité d'un Cadre Mathématique Unifié

L'extension à plusieurs modalités a révélé les limites des **architectures** de type SOM isolée ou réseau associatif spécialisé. De multiples travaux ont souligné qu'on avait besoin d'un **réseau** davantage "libre" : au lieu de cloisonner chaque flux dans un module (une carte), il serait plus fécond de concevoir un unique **ensemble** d'entités, au sein duquel n'importe quelle paire (ou sous-ensemble) pourrait maintenir une liaison  $\omega_{i,j}$  reflétant la "synergie"  $S(\mathcal{E}_i, \mathcal{E}_j)$ . Cette vision implique un **espace d'interaction** où la **pondération** entre entités évolue au fil du temps, en s'appuyant sur des règles locales de renforcement ou d'affaiblissement. Le concept est proche du **Synergistic Connection Network (SCN)** décrit dans la suite du **DSL** :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta[S(\mathcal{E}_i, \mathcal{E}_j) - \tau \omega_{i,j}(t)].$$

Ici,  $\mathcal{E}_i$  et  $\mathcal{E}_j$  peuvent être des entités de natures diverses (sensorielles, symboliques, etc.), et la fonction  $S$  se charge d'estimer le degré de "**cohérence**" ou de "**complémentarité**".

### D. Évolution Vers la Pondération Adaptative

Dans les premières "SOM doubles" ou architectures robotiques multimodales, les liens entre modalités restaient parfois figés ou suivaient un schéma de **mapping** imposé. Les itérations ultérieures, plus ambitieuses, ont cherché à rendre ces **connexions** elles-mêmes **dynamiques**, comme s'il s'agissait d'une extension de la logique de Hebb ou de la compétition neuronale. On posait donc la question :

"Pourquoi ne pas autoriser la liaison *inter-cartes* à se mettre à jour en temps réel, au même titre que les liaisons *intra-carte* ?"  
 Cette interrogation a mené à l'idée d'un **réseau unique**, plutôt que de modules encapsulés, préparant la conception que le **Deep Synergy Learning** développera : un SCN global, évoluant au gré de la synergie perçue entre tous les **types** d'entités.

### E. Préparation du Terrain pour le DSL

En repensant l'architecture de l'apprentissage auto-organisé sous un angle plus universel, on arrivait progressivement à un **paradigme** où l'on n'avait plus à distinguer a priori les données en tant que "vision", "tactile" ou "auditive". On leur attribuait plutôt un statut générique d'**entités** dans un réseau, reliées par des liaisons qui s'ajustaient en fonction d'un **score** de co-opération ou de co-information. Ainsi se dessinait déjà l'idée d'une **synergie** non supervisée, en suivant le fil rouge des premiers prototypes multimodaux et en l'étendant à un cadre totalement **auto-adaptatif**.

Les travaux de recherche dans la robotique sensorielle et l'intégration multimodale ont de fait affirmé la  **faisabilité** d'une telle approche. Ils ont mis en évidence que la conclusion "ces deux flux s'avèrent souvent synchronisés, donc ils doivent être reliés et codés conjointement" pouvait être atteinte **sans** recours à un signal d'erreur supervisé. C'est sur cette base qu'ont émergé des pistes ultérieures visant une formalisation plus large de la "synergie" et, en particulier, l'essor de

paradigmes comme le **Deep Synergy Learning** qui généralisent ces principes en offrant un langage pour la **coopération** entre entités et la **réorganisation** des liaisons  $\omega_{i,j}$ .

Dès lors, la volonté de **fusionner** plusieurs types de données a joué le rôle d'un catalyseur, forçant la communauté à sortir d'architectures préétablies pour considérer un **réseau unifié** et malléable. Les règles d'évolution des pondérations sont alors guidées par une fonction  $S(i,j)$  – future “fonction de synergie” – et cette logique d'interactions adaptatives reste au cœur du **DSL**.

En conclusion, ces initiatives multimodales ont poussé à systématiser l'**auto-organisation** au-delà de la simple proximité de vecteurs, afin d'engendrer un **cadre** où toute forme de similarité ou de **complémentarité** (visuelle, tactile, auditive, symbolique) puisse s'exprimer sous forme de synergie, ouvrant ainsi la voie aux principes fondateurs du **Deep Synergy Learning**.

### *2.1.5. Origine du DSL et Ses Premiers Manifestes*

Les divers courants présentés dans les sections précédentes (2.1.1 à 2.1.4) — allant des travaux pionniers en auto-organisation jusqu'aux tentatives de fusion multimodale — ont fini par converger vers l'idée de constituer un **cadre unifié** où chaque entité (capteur, neurone, module symbolique, etc.) pourrait interagir avec les autres via un mécanisme de **synergie adaptative**. C'est dans ce contexte qu'est apparue la notion de **Deep Synergy Learning (DSL)**, que je souhaitais formaliser et pousser plus loin la dynamique de l'**auto-organisation** pour qu'elle englobe l'hétérogénéité des flux (multimodaux, symboliques, etc.) tout en restant fidèle à la tradition d'un apprentissage non supervisé. Le sous-chapitre qui suit (2.1.5) s'intéresse à la genèse historique et à la cristallisation de l'étiquette “DSL” dans la littérature, puis à la formalisation progressive d'un “Synergistic Connection Network (SCN)”.

#### **2.1.5.1. Quand et Comment l'Étiquette “Deep Synergy Learning” Apparaît pour la Première Fois : la Publication dans cet Ouvrage**

Au tournant des années 2010, divers laboratoires et chercheurs ont exploré l'idée de **réseaux adaptatifs** capables de gérer simultanément des entités ou des modalités multiples, dans la lignée des inspirations décrites dans les sections précédentes. Toutefois, le terme “**Deep Synergy Learning (DSL)**” tel qu'il est présenté ici n'avait jusqu'alors **jamais** fait l'objet d'une publication ou d'une formalisation officielle. C'est dans **ce livre**, pour la première fois, que je rends publiques et détaillées les bases conceptuelles et le fonctionnement d'un **réseau** apte à intégrer la notion de **synergie** au sens large, qu'il s'agisse d'entités perceptuelles, symboliques, ou mixtes.

L'idée de la **synergie** m'accompagne depuis plus de **quarante ans**, influencée par mes travaux initiaux en informatique, par les modèles d'**auto-organisation** neuronale (type Hopfield, Kohonen) et par la **thermodynamique hors équilibre**. Au fil des décennies, ces réflexions se sont enrichies de la logique “hebbienne”, des simulations biologiques et de la fusion multimodale, mais n'ont jamais été diffusées dans un ouvrage ou un article de référence. Aucun document antérieur ne porte l'étiquette “**Deep Synergy Learning**” au sens où je l'entends aujourd'hui. Ce livre constitue dès lors la **première** présentation formalisée du DSL, énonçant ses principes, ses justifications théoriques et ses applications potentielles.

On trouvera dans les chapitres ultérieurs la description du **Synergistic Connection Network (SCN)**, pièce maîtresse du DSL, ainsi que l'ensemble des règles de mise à jour permettant de faire

évoluer les liaisons  $\omega_{i,j}$  selon une fonction de **synergie**  $S(i,j)$ . Si l'on retrouve des traces d'inspirations diverses (réseaux associatifs, cartes topologiques, plasticité synaptique), la synthèse qui en résulte se veut inédite : elle généralise l'**auto-organisation** à un large spectre d'entités, tout en recherchant un ordre émergent via des **pondérations adaptatives**. Cette publication, en franchissant pour la première fois les frontières des laboratoires et des notes internes, aspire donc à faire connaître le DSL et à favoriser son adoption dans un grand éventail de contextes, de la robotique sensorielle à la cognition symbolique.

Ainsi, le **Deep Synergy Learning**, dont je pose ici les fondements, n'avait jusque-là jamais été officiellement exposé ni dans un livre ni dans un article. Ce texte inaugure donc son entrée dans la **littérature** scientifique, concrétisant une vision mûrie de longue date mais demeurée confidentielle jusqu'à présent.

### 2.1.5.2. Évolution vers la Formalisation d'un "SCN" (Synergistic Connection Network)

La **naissance** du **Deep Synergy Learning (DSL)** s'est accompagnée d'une volonté de conférer à la **synergie adaptative** une **incarnation concrète** au sein d'un réseau à la fois généraliste et dynamique. Les réseaux associatifs traditionnels (Hopfield) et les cartes topologiques (SOM) ont inspiré certains principes, mais ils restaient marqués par des limites : absence de plasticité continue, champ d'application restreint, difficulté à intégrer des entités non neuronales ou des flux multimodaux. Le **SCN**, au contraire, se veut un **réseau** entièrement **évolutif**, où chaque pondération  $\omega_{i,j}$  dépend d'une fonction de synergie  $S(i,j)$  susceptible d'unifier différents types d'entités (capteurs sensoriels, modules symboliques, etc.). Les paragraphes ci-dessous (2.1.5.2.1 à 2.1.5.2.5) retracent la **progression** qui a permis d'aboutir à ce cadre.

#### *A. Passer d'un Réseau "Fixe" à un Réseau "Entièrement Dynamique"*

Les approches antérieures telles que les **SOM** de Kohonen ou les **réseaux** Hopfield présentaient un fonctionnement essentiellement **statique**. Une SOM impose une grille et un mode de voisinage prédéterminé, tandis que les poids d'un réseau Hopfield sont généralement calculés une fois pour toutes sur la base de motifs ciblés. Ces méthodes ne permettent pas de gérer en continu l'arrivée de nouvelles entités ou de mettre à jour les liaisons dans des environnements en transformation permanente. Inspiré par la **plasticité neuronale**, j'ai suggéré que le réseau pourrait évoluer par itérations successives, les pondérations  $\omega_{i,j}(t)$  se renforçant ou s'atténuant selon un **score de synergie**  $S(i,j)$ . L'équation générique (développée plus tard dans ce livre) concrétise cette idée en intégrant un **terme de renforcement** (quand la synergie est élevée) et un **terme de décroissance** (pour éviter l'emballement), assurant un équilibre entre stabilité et adaptabilité.

#### *B. Notion de "Synergistic Connection Network"*

Sous l'impulsion de ces réflexions, est né le concept de **Synergistic Connection Network (SCN)**, un graphe (voire un hypergraphe) dont les nœuds correspondent à des **entités**  $\mathcal{E}_1, \mathcal{E}_2, \dots$ , alors que les arêtes pondérées  $\omega_{i,j}$  incarnent la liaison entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . La distinction cruciale, par rapport aux architectures classiques, réside dans le fait que ces liaisons ne sont pas figées : elles varient sous

l'effet d'une **fonction**  $S(\mathcal{E}_i, \mathcal{E}_j)$  évaluant la synergie. À chaque étape (itération, nouvelle donnée, flux continu), les pondérations se recalculent pour refléter le degré de coopération ou de similitude actuelle entre les entités. Ce **réseau vivant** autorise la découverte de structures auto-organisées dans des champs aussi divers que la **fusion sensorielle**, la **cohérence** symbolique ou la **mise en correspondance** de flux d'origine variée.

### C. Dynamique de la Fonction $\omega_{i,j}(t)$ et Synergie $S(i, j)$

Le **SCN** s'appuie sur un mécanisme de **plasticité locale** où la pondération  $\omega_{i,j}(t)$  entre deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  obéit, à chaque itération, à une équation de type

$$\omega_{i,j}(t + 1) = \omega_{i,j}(t) + \eta[S(\mathcal{E}_i, \mathcal{E}_j) - \tau \omega_{i,j}(t)].$$

Ici,  $\eta$  est un **taux d'apprentissage**, tandis que  $\tau$  joue le rôle d'un **coefficient** de régulation, souvent nommé "terme de décroissance". La **fonction**  $S(i, j)$  peut être une simple corrélation ou quelque chose de plus élaboré : distance inversée, similarité cosinus, information mutuelle, etc. Cette souplesse correspond à l'ambition du SCN de **fusionner** différents types de données, en adoptant une métrique ou une fonction de synergie appropriée pour chaque couple (ou sous-groupe) d'entités. L'effet global est un **réseau** qui se reconfigure sans cesse, renforçant les liaisons cohérentes et affaiblissant les autres, selon les principes hebbiens et les idéaux de la **physique hors équilibre** (cf. chapitres 2.1.3.1 et 2.1.3.2).

### D. Diffusion dans des Publications et Ateliers

Au fil des travaux, ce schéma évolutif a pris forme dans des **rapports** et **communications** spécialisées, notamment dans le cadre de congrès sur l'**apprentissage non supervisé**. Les premières applications pratiques concernaient la fusion multisensorielle ou la classification évolutive, montrant que le SCN sait gérer la venue de nouvelles données, adapter les pondérations au fil de l'expérience et former des **clusters** persistants. Parmi les questions soulevées figurent la **scalabilité** (comment se comporte le SCN avec un grand nombre d'entités ?) et la **stabilité** (comment éviter que les pondérations ne divergent ou ne s'étiolent toutes ?). Ces interrogations ont orienté la mise au point de mécanismes de **parsimonie** (seuils, inertie minimale), d'**inhibition** compétitive et de **taux d'apprentissage** dynamique, dont les détails seront explicités dans les prochains chapitres.

### E. Positionnement pour la Suite de l'Ouvrage

Ainsi, la création du **Synergistic Connection Network** marque une étape décisive dans la concrétisation du **Deep Synergy Learning**. Pour la première fois, un **réseau** est conçu pour embrasser la diversité des entités (capteurs de nature variée, modules symboliques, etc.), tout en veillant à ce que les liaisons  $\omega_{i,j}$  restent **adaptatives** et que la **synergie**  $S(i, j)$  puisse se décliner sous des formes diverses. Les chapitres suivants de cet ouvrage (chapitres 3, 4, 5) approfondiront la formulation analytique, les algorithmes de mise à jour, et les fondements mathématiques de la



dynamique  $\omega_{i,j}(t)$ . On y détaillera notamment la notion de **cluster** en tant qu'attracteur emergent, la gestion de la **parsimonie** afin de maintenir la lisibilité et la scalabilité, ainsi que les méthodes pour **évaluer** la pertinence des synergies dans des scénarios multimodaux réels.

En définitive, la formalisation du **SCN** illustre la transition d'une simple **idée** (renforcer les liens lorsqu'on détecte une forte co-occurrence) à un **cadre** complet d'**auto-organisation** appliqué aux entités hétérogènes. Le DSL s'appuie désormais sur cette armature conceptuelle pour proposer un système de pondérations locales, à la fois général et flexible, ouvrant la voie à une **intelligence** non supervisée, capable de gérer la diversité des données et de s'auto-structurer en conséquence.

### 2.1.5.3. Lien Direct avec la Suite du Livre : Axes Théoriques Restés Inexplorés

La **genèse** du **Deep Synergy Learning (DSL)** et la formalisation du **Synergistic Connection Network (SCN)**, évoquées dans les sections précédentes, laissent entrevoir un champ encore **vaste** de problématiques que la littérature n'a pas couvertes de manière exhaustive. Le présent ouvrage se propose précisément de combler plusieurs de ces **lacunes** et de consolider les bases théoriques et pratiques du DSL. La discussion suivante met en évidence les principaux **axes** de recherche inachevés, tout en exposant la manière dont les chapitres ultérieurs (3 à 15) s'emploient à les approfondir ou à proposer des réponses partielles.

#### A. Extension de la Fonction de Synergie à des Contextes Plus Complexes

La plupart des modèles initiaux se limitaient à des **fonctions** de similarité relativement simples, comme la distance euclidienne ou la corrélation binaire, pour calculer la synergie entre deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . Or, certains domaines exigent une **cohérence** ou une **interaction** plus élaborée, qui peut s'appuyer sur des notions de co-information n-aire ou sur des métriques conditionnelles sophistiquées. D'un point de vue mathématique, on se heurte alors à la nécessité de définir

$$S(\mathcal{E}_1, \mathcal{E}_2, \dots, \mathcal{E}_n)$$

pour un sous-groupe de taille  $n > 2$ , ce qui amène à de nouveaux défis de calcul et de stockage. Le **Chapitre 3** aborde la représentation des entités et jette les bases de la synergie, tandis que le **Chapitre 12** traite spécifiquement de la **synergie n-aire**, en exposant des formules permettant de saisir les interactions multivariées.

Par ailleurs, le **DSL** se révèle apte à gérer des flux en temps continu. Cela suppose que la fonction  $S(i, j)$  demeure **adaptative** au fil du temps, afin de prendre en compte l'évolution possible des entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . Les **Chapitres 9** et **11**, consacrés respectivement aux scénarios temps réel et à la robustesse/sécurité, examinent la façon dont le **SCN** peut conserver la mémoire des synergies passées, tout en s'ajustant à l'arrivée de nouvelles données ou d'entités inédites.

#### B. Contrôle de la Stabilité et Algorithmes d'Optimisation

Il est légitime de s'interroger sur la **stabilité** de la règle de mise à jour

$$\omega_{i,j}(t + 1) = \omega_{i,j}(t)\eta[S(i, j) - \tau \omega_{i,j}(t)].$$

D'un point de vue analytique, il s'agit de savoir si ces équations peuvent engendrer des **oscillations**, ou même des comportements chaotiques, lorsque le réseau s'agrandit ou que la matrice  $\{\omega_{i,j}\}$  atteint une dimension élevée. Les **Chapitres 4, 5, 7 et 10** explorent ces problématiques. Les solutions proposées incluent, entre autres, l'ajout d'un mécanisme d'**inhibition compétitive** (destiné à empêcher l'explosion d'un trop grand nombre de liaisons fortes), l'introduction de seuils minimaux pour couper les connexions trop faibles ou encore l'élaboration de stratégies de **parsimonie** structurelle.

La **scalabilité** en présence de milliers, voire de millions, d'entités devient un enjeu majeur. Les approches naïves peuvent rapidement se heurter à une complexité en  $O(n^2)$ . Le **Chapitre 7** détaille des algorithmes plus performants, tels que l'**échantillonnage** ou l'**approximation** de voisinage, et puise dans des analogies avec la **physique statistique** (par exemple le recuit simulé) afin de maintenir la convergence sans tout calculer explicitement. Les utilisateurs du **DSL** se voient ainsi offrir un éventail de méthodes pour contrôler la densité du réseau et éviter que les liaisons  $\omega_{i,j}$  ne prolifèrent ou ne s'effondrent de manière incontrôlée.

### *C. Ouverture vers la Multimodalité Avancée et le Raisonnement Symbolique*

Un autre front de recherche concerne la **multimodalité** lorsqu'elle est poussée à l'extrême : il ne s'agit plus seulement de relier deux flux (vision + tactile), mais d'intégrer simultanément des sources diverses (images, sons, textes, logique symbolique). Dans cette optique, on souhaite s'assurer que la **fonction de synergie**  $S(\cdot, \cdot)$  ou  $S(\cdot, \cdot, \cdot)$  soit assez flexible pour gérer des écosystèmes d'entités hétérogènes, tout en garantissant que le SCN ne se fragmente pas en une myriade de **clusters** trop spécialisés et déconnectés. Le **Chapitre 8** se penche spécifiquement sur ces scénarios multimodaux et propose des pistes pour favoriser la **transversalité** entre flux distincts. Les **Chapitres 14 et 15**, davantage orientés applications, reviennent sur divers cas concrets (fusions de flux audio-visuel-textuel, organisation de connaissances dans un cadre hybride).

La présence d'**entités symboliques** (règles logiques, axiomes) étend encore le périmètre d'action du DSL, et il s'agit alors de caractériser la **synergie** entre une entité perceptuelle (un capteur) et une entité abstraite (une proposition symbolique). Les **Chapitres 5, 13 et 14** abordent cette problématique en discutant des approches dites **neuro-symboliques**, où un réseau gère simultanément de la "matière" sub-symbolique (poids, neurones virtuels) et des "objets" symboliques. La recherche sur la cohabitation de ces deux mondes reste néanmoins incomplète, et le livre ne prétend pas en livrer une théorie achevée, mais propose des esquisses d'**implémentation** et des modèles hybrides encourageant un raisonnement distribué.

### *Conclusion sur les Axes Théoriques Restés Inexplorés*

Le **Deep Synergy Learning**, formellement présenté dans ce livre pour la première fois (cf. section 2.1.5.1), débouche sur un **cadre** suffisamment large pour accueillir des entités variées et autoriser une dynamique de pondérations adaptatives. Les **lacunes** ou **chantiers** toujours ouverts concernent d'abord l'élaboration de **fonctions de synergie** toujours plus polyvalentes (synergie n-aire, co-information conditionnelle). Ils portent aussi sur l'analyse **mathématique** de la stabilité du **SCN** (en particulier dans les grands réseaux), et enfin sur la **fusion** multimodale avancée associant données sensorielles, logiques symboliques et autres ressources. Les chapitres à venir s'emploient à clarifier chacun de ces thèmes :

- Les **fondements mathématiques** (Chapitre 3 et Chapitre 4)



- Les **règles de mise à jour** et la **structure interne** du SCN (Chapitre 5, Chapitre 7)
- Les scénarios d'**évolution en temps réel** (Chapitre 9) et de **multimodalité** (Chapitre 8, Chapitre 14)
- Les **cas pratiques** et les extensions potentielles (Chapitre 15)

Ainsi, le livre fournit un socle formel pour la **synergie adaptative** entre entités. La variété des pistes encore ouvertes souligne combien le **DSL** n'est pas un aboutissement final, mais plutôt un **cadre évolutif** lui-même, évoluant au rythme des investigations en intelligence artificielle, en neurosciences computationnelles et en physique des systèmes complexes.

### *Conclusion sur les Axes Théoriques Restés Inexplorés*

En clair, si la **genèse** du Deep Synergy Learning et la mise en place du SCN marquent une avancée notable dans l'histoire de l'auto-organisation, plusieurs **chantiers** théoriques et pratiques restent à développer :

- **Formalisation** complète d'une synergie adaptée à la cohabitation de multiples types d'entités (sub-symboliques, symboliques) ;
- **Étude** approfondie de la **stabilité** et des attracteurs dans un réseau dont les pondérations évoluent en continu, notamment en présence de synergie n-aire ;
- **Optimisation** et **scalabilité**, cruciales pour supporter un grand nombre d'entités dans le SCN ;
- **Transitions** vers la multimodalité avancée et la prise en compte d'un raisonnement symbolique plus élaboré.

Les chapitres suivants du livre aborderont bon nombre de ces **challenges**, proposant des embryons de solution ou montrant des applications concrètes. Le lecteur verra comment ces problématiques s'imbriquent dans la dynamique du SCN, et comment l'héritage historique présenté ici (depuis les réseaux associatifs, les cartes topologiques, jusqu'aux modèles multi-sensoriels et la physique statistique) trouve sa continuité dans un cadre unifié **Deep Synergy Learning**.

## 2.2. Principes Mathématiques de Base

Le **Deep Synergy Learning (DSL)** s'appuie sur un ensemble de **principes mathématiques** permettant de définir formellement les entités (leurs caractéristiques) et les règles d'**auto-organisation** du réseau. Contrairement à certains paradigmes neuronaux classiques (où l'on manipule seulement des poids fixes ou un gradient d'erreur), le DSL met en avant la **fonction de synergie**  $S$  et la dynamique de mise à jour des pondérations  $\omega_{i,j}$ . Ces concepts reposent sur des bases analytiques qui font intervenir la notion d'**espace de représentation**, de **similarité** ou de **co-information**, et de **procédés d'évolution** (discrets ou continus). Le présent chapitre (2.2) en présente les **fondements**, en plusieurs sous-sections :

- **2.2.1.** Définition des Entités et de la Fonction de Synergie
- **2.2.2.** Mise à Jour des Pondérations : Formule Générale
- **2.2.3.** Règles de Parsimonie et Seuil de Connexion
- **2.2.4.** Notions d'États Internes et Auto-Organisation
- **2.2.5.** Exemples Illustratifs

Ce sont les **piliers** mathématiques sur lesquels s'édifie le **Synergistic Connection Network (SCN)** et, plus généralement, l'ensemble de la logique d'**auto-organisation** du DSL.

### 2.2.1. Définition des Entités et de la Fonction de Synergie

La première étape pour tout modèle DSL consiste à **identifier** les **entités**  $\{\mathcal{E}_1, \dots, \mathcal{E}_n\}$  qui composeront le réseau et, surtout, à spécifier la **fonction de synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$ . Si la littérature sur l'apprentissage non supervisé (SOM, Hopfield, etc.) imposait en général un format vectoriel unique, le DSL se distingue par sa capacité à gérer des **représentations hétérogènes** (capteurs, flux symboliques, extraits d'un transformeur, etc.). Les sections suivantes (2.2.1.1 à 2.2.1.3) détaillent cette notion de **définition d'entités** et la façon dont on conçoit la **fonction**  $S$ .

#### 2.2.1.1. Rappel : Une Entité $\mathcal{E}_i$ (Vecteur de Caractéristiques, Représentation Symbolique, etc.)

Le **Deep Synergy Learning (DSL)** s'articule autour de la notion d'**entité**  $\mathcal{E}_i$  et de la **synergie** définie sur l'ensemble de ces entités. Dans la pratique, on considère que chaque  $\mathcal{E}_i$  fait partie d'un **ensemble**  $\mathcal{X}$  qui peut prendre des formes variées : espace vectoriel, structure symbolique, ou représentation probabiliste. Il est indispensable de disposer d'une **mesure** ou d'une **similarité** pour comparer deux entités, car c'est cette comparaison qui sous-tend la **pondération**  $\omega_{i,j}$  et la **dynamique** d'auto-organisation du DSL. Les développements ci-après soulignent les différents cas de figure envisageables (sections vectorielles, symboliques, probabilistes) et introduisent un formalisme plus général permettant de prendre en compte des **espaces** hétérogènes.

### A. Entités dans un Espace Vectoriel $\mathbb{R}^d$

Un premier cas, souvent le plus courant, suppose que chaque entité  $\mathcal{E}_i$  est représentée par un **vecteur** réel de dimension  $d$ . On écrit alors

$$\mathcal{E}_i \in \mathbb{R}^d,$$

et on dispose de **distances** ou de **similarités** naturelles : distance euclidienne, distance Minkowski, similarité cosinus, etc. Les propriétés de l'analyse vectorielle facilitent la définition de mesures telles que

$$d(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\| \quad \text{ou} \quad \text{sim}(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\|\mathbf{x}_i\| \|\mathbf{x}_j\|}.$$

Cette configuration se rencontre fréquemment dans la pratique (traitement d'images, de séries temporelles, etc.), et fournit une base conceptuelle aisée pour **construire** la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  par inversion de distance ou par évaluation de similarité, prolongeant l'idée de la corrélation ou de la co-occurrence.

### B. Structures Symboliques ou Espaces Discrets

Une **entité**  $\mathcal{E}_i$  peut aussi correspondre à un **concept** symbolique, un **graphe** local, ou tout autre objet appartenant à un **espace** discret où aucune structure vectorielle n'est immédiatement disponible. Les ensembles de cette nature exigent la définition d'une **mesure** ou d'une **similarité** plus adaptée, prenant la forme d'une distance combinatoire, d'une ressemblance sémantique, ou d'une co-information symbolique. Formellement, on peut écrire

$$\mathcal{E}_i \in \mathcal{X}_{\text{symbolique}},$$

et introduire une fonction  $\text{sim}(\mathcal{E}_i, \mathcal{E}_j)$  prenant des valeurs dans  $[0,1]$  ou  $\mathbb{R}^+$ . Ce cadre est crucial dans le DSL, qui autorise la **co-existence** d'entités hétérogènes et nécessite donc que la **synergie** s'applique aussi à des représentations non purement numériques. Par exemple, on peut concevoir une mesure de similarité symbolique  $\sigma(\mathcal{E}_i, \mathcal{E}_j)$  qui vaut 1 si deux symboles sont identiques, ou prend une valeur intermédiaire si l'on recourt à une **ontologie** reliant les concepts.

### C. Espace de Probabilité ou Représentation Probabiliste

Dans certains travaux, l'entité  $\mathcal{E}_i$  est un **modèle probabiliste**, par exemple une **distribution**  $p_i(x)$ . On peut alors définir la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  par une divergence ou une similarité adaptée, telles que la **KL divergence** ou la **distance Wasserstein**. Ainsi, on écrit

$$\mathcal{E}_i \in \mathcal{P}(\mathcal{X}),$$

où  $\mathcal{P}(\mathcal{X})$  désigne l'ensemble des distributions sur un espace  $\mathcal{X}$ . La fonction  $S$  peut alors prendre des formes comme

$$S(\mathcal{E}_i, \mathcal{E}_j) = \exp(-d_{\text{KL}}(p_i, p_j)) \quad \text{ou} \quad S(\mathcal{E}_i, \mathcal{E}_j) = \text{Wasserstein}(p_i, p_j),$$

autorisant l'auto-organisation de **modèles** statistiques sans référence explicite à des labels supervisés.

#### D. Forme Générale et Hypothèses Topologiques

On adopte volontiers la notation

$$\mathcal{E}_i \in \mathcal{X},$$

où  $\mathcal{X}$  est un **espace topologique** (ou mesurable), muni d'une **distance** ou d'une **similarité**. Le **triplet**  $(\mathcal{X}, d, \mu)$  ou  $(\mathcal{X}, S, \mu)$  permet de considérer soit des distances classiques, soit des “**fonctions de synergie**” plus directes. Le **DSL** n'exige pas nécessairement que les entités se situent dans un unique espace vectoriel : différentes **sous-familles** d'entités peuvent coexister, chacune dotée de sa propre mesure.

#### E. Propriétés Clés : Séparabilité, Embedding Vectoriel, Extension Symbolique

Certaines **propriétés** mathématiques facilitent l'étude de la synergie. Par exemple, une **distance**  $d$  sur  $\mathcal{X}$  procure un espace métrique complet, ce qui autorise une **analyse** de la convergence de la fonction de synergie si elle se base sur  $d$ . Sous des conditions favorables (e.g. la structure de  $\mathcal{X}$  est hilbertienne), il est possible d'**implanter**  $(\mathcal{X}, d)$  dans un espace  $\mathbb{R}^d$  ou un **espace de dimension infinie** (via un **noyau**), ouvrant la voie à des similarités usuelles (produit scalaire, gaussien, cosinus). Lorsque  $\mathcal{X}$  est un ensemble **discret** (fin ou dénombrable), on peut définir une **similarité**  $\sigma(\mathcal{E}_i, \mathcal{E}_j)$  dans  $[0,1]$  basée sur une codification combinatoire ou sémantique.

#### F. Gestion de l'Hétérogénéité des Entités

Le **DSL** poursuit un objectif d'**intégration** de différents types d'entités  $\{\mathcal{E}_1, \dots, \mathcal{E}_n\}$ , qui peuvent appartenir à divers espaces  $\mathcal{X}_1, \mathcal{X}_2, \dots, \mathcal{X}_k$ . Pour comparer  $\mathcal{E}_i \in \mathcal{X}_p$  et  $\mathcal{E}_j \in \mathcal{X}_q$ , il est indispensable de **prolonger** les mesures ou d'introduire un **mécanisme de passerelle** (embedding commun, correspondance symbolique...). C'est dans cette optique qu'est formulé le **théorème** sur la consistance de la synergie inter-espaces, garantissant l'existence d'une fonction globale  $\Sigma$  unifiant les similarités  $\sigma_p$  dans un unique référentiel. Les détails topologiques (réunion disjointe, mise en place d'une distance globale  $\delta$  séparant strictement les blocs) assurent la **continuité** de la fonction  $\Sigma$ .

#### Exemple de Théorème (Consistance de la Synergie Inter-Espaces)

##### Enoncé (informel).

Soient  $\{\mathcal{X}_m\}_{m=1}^k$  des espaces (topologiques ou mesurables), chacun muni d'une fonction de similarité  $\sigma_m$ . Sous hypothèses de continuité uniforme et de séparation stricte entre blocs, il existe une **extension**  $\Sigma$  définie sur la réunion disjointe  $\mathcal{U} = \sqcup_{m=1}^k (\mathcal{X}_m \times \{m\})$ , telle que

$$\Sigma|_{\mathcal{X}_m \times \mathcal{X}_m} = \sigma_m \quad \text{et} \quad \Sigma((x, p), (y, q)) = c_{p,q} \quad (p \neq q).$$

On obtient ainsi une fonction  $\Sigma: \mathcal{U} \times \mathcal{U} \rightarrow [0,1]$  (ou  $\mathbb{R}^+$ ), **uniformément continue** sur la distance  $\delta$  définie par bloc, offrant un cadre d'unification pour l'**auto-organisation** dans un seul **Synergistic Connection Network (SCN)**.

La démonstration s’appuie sur la **réunion disjointe** et la définition d’une distance globale  $\delta$  qui maintient une séparation stricte ( $\Delta_0$ ) entre blocs  $\mathcal{X}_p$  et  $\mathcal{X}_q$  pour  $p \neq q$ . La construction assure que  $\Sigma$  est **intra-bloc** égale à  $\sigma_m$ , et **inter-bloc** égale à une constante (souvent zéro). On en déduit que  $\Sigma$  est **uniformément continue**, ce qui garantit la possibilité de réaliser une **analyse** de stabilité ou de convergence si l’on souhaite faire évoluer un réseau global.

### Conclusion (2.2.1.1)

Au terme de cette analyse, on retient que, dans le **Deep Synergy Learning**, chaque **entité**  $\mathcal{E}_i$  peut être un simple **vecteur** (cas fréquent en  $\mathbb{R}^d$ ), une **structure symbolique** (concept discret, graphe), ou même un **modèle probabiliste**. L’important est de disposer d’une **fonction** (distance, similarité, co-occurrence) permettant de quantifier la “**proximité**” ou la “**coopération**” entre entités, sans quoi la notion de **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  reste vide. Dans les chapitres suivants, on verra **comment** décliner cette synergie (section 2.2.1.2, centrée sur distance, corrélation, information mutuelle) et **comment** l’utiliser pour piloter la **mise à jour** des pondérations  $\omega_{i,j}$  (section 2.2.2). Ainsi, le **DSL** se veut un cadre assez large pour **unifier** la comparaison d’entités de natures diverses et lancer, sur cette base, un **SCN** (Synergistic Connection Network) unique, où les liens évoluent selon la valeur de la synergie détectée.

### 2.2.1.2. Qu’est-ce que $S(\mathcal{E}_i, \mathcal{E}_j)$ ? Exemples (distance, similarité, co-information)

Pour décrire la **synergie** reliant deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  dans le **Deep Synergy Learning (DSL)**, on introduit une **fonction**  $S(\mathcal{E}_i, \mathcal{E}_j)$  qui, pour chaque couple d’entités, mesure leur **coopération** ou leur **affinité**. On formalise souvent ce concept par

$$S: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^+,$$

où  $\mathcal{X}$  représente l’espace (ou la **réunion** d’espaces) hébergeant les entités  $\mathcal{E}_i, \mathcal{E}_j$ . Selon la **nature** de  $\mathcal{X}$  et les **objectifs** (clustering, fusion multimodale, etc.), la fonction  $S$  peut être fondée sur différents concepts (distance, similarité, information mutuelle). Les développements ci-après (2.2.1.2.1 à 2.2.1.2.3) exemplifient trois grandes **familles** de mesures couramment mises en avant dans le DSL.

#### A. Distance

Lorsque les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  appartiennent à un **espace métrique**  $(\mathcal{X}, d)$ , on peut définir la synergie comme une **fonction** décroissante de la distance :

$$S(\mathcal{E}_i, \mathcal{E}_j) = f(d(\mathcal{E}_i, \mathcal{E}_j)),$$

avec, par exemple,  $f(x) = 1/(1+x)$  ou  $f(x) = \exp(-\alpha x)$ . L’intuition repose sur l’idée qu’une faible **distance** implique une **coopération** plus marquée. Dans un cadre vectoriel  $\mathbb{R}^d$ , si l’on prend la distance euclidienne

$$d(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|,$$

on peut fixer

$$S(\mathcal{E}_i, \mathcal{E}_j) = \exp(-\alpha \|\mathbf{x}_i - \mathbf{x}_j\|),$$

de sorte que deux entités **proches** dans  $\mathbb{R}^d$  obtiennent un score  $S$  élevé, et deux entités **éloignées** un score proche de zéro. Sur le plan des **propriétés**, la distance garantit la **symétrie**  $S(\mathcal{E}_i, \mathcal{E}_j) = S(\mathcal{E}_j, \mathcal{E}_i)$  si le choix de la fonction  $f$  respecte cette invariance.

### B. Similarité ou Corrélation

Une autre façon de concevoir la **synergie** privilégie l'idée de **similarité** ou de **corrélation** statistique. Par exemple, si  $\mathcal{E}_i$  et  $\mathcal{E}_j$  sont deux vecteurs  $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^d$ , on peut recourir à la **similarité cosinus** :

$$S(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\|\mathbf{x}_i\| \|\mathbf{x}_j\|}.$$

Dans ce cas, plus l'**angle** entre les vecteurs est faible, plus la valeur de  $S$  est grande. Pour des variables **aléatoires**  $X, Y$ , on peut choisir le **coefficient de corrélation**  $\rho(X, Y)$ , par exemple la **corrélation de Pearson**, ou toute autre mesure similaire. D'un point de vue algorithmique, le **DSL** exploitera alors ces scores comme un indicateur de synergie : lorsque deux entités se révèlent linéairement (ou faiblement) interdépendantes, leur lien  $\omega_{i,j}$  se renforce modérément ; lorsqu'elles affichent une corrélation élevée, le **renforcement** sera marqué. Cette logique, conforme aux paradigmes associatifs (Hopfield) ou hebbiens, consolide l'idée que la **co-activation** façonne le **réseau** en intensifiant les liaisons utiles.

### C. Information Mutuelle ou Co-Information

Dans des situations où les entités  $\mathcal{E}_i, \mathcal{E}_j$  représentent des **variables aléatoires** (ou des distributions de probabilité), une approche consiste à mesurer la **synergie** via l'**information mutuelle** :

$$S(\mathcal{E}_i, \mathcal{E}_j) = I(\mathcal{E}_i; \mathcal{E}_j) = \int p(x, y) \log[p(x, y)/p(x)p(y)] dx dy.$$

Cette grandeur quantifie la **dépendance** entre deux variables : plus l'une réduit l'incertitude sur l'autre, plus leur information mutuelle est élevée. Dans un cadre plus général, on parle de **co-information**  $I(\mathcal{E}_1; \mathcal{E}_2; \dots; \mathcal{E}_k)$  lorsque plusieurs variables interagissent collectivement (voir Chapitre 12). Sur le plan **théorique**, l'information mutuelle a l'avantage de capturer des dépendances **non linéaires**, dépassant la simple corrélation. Si  $S(\mathcal{E}_i, \mathcal{E}_j)$  est défini comme  $I(\mathcal{E}_i, \mathcal{E}_j)$ , alors la valeur s'annule en cas d'indépendance et s'accroît avec la force de la dépendance. Dans le **DSL**, ce schéma se prête bien à un **renforcement** de  $\omega_{i,j}$  lorsque les variables s'avèrent interdépendantes.

Toutefois, la mise en pratique de l'information mutuelle peut exiger des **estimations** ou des **approximations** si l'on ne connaît pas explicitement les distributions  $p(x)$ . Des estimateurs basés

sur des techniques de **binning**, de **k plus proches voisins** ou des méthodes paramétriques (ex. gaussiennes) sont souvent employés. Les performances et la fiabilité de ces estimateurs conditionnent alors la précision de la fonction  $S$ .

### *Conclusion sur le Choix de la Fonction de Synergie*

La décision quant à la fonction  $S(\mathcal{E}_i, \mathcal{E}_j)$  oriente la **dynamique** du **DSL** : seules les paires  $\mathcal{E}_i, \mathcal{E}_j$  présentant un score élevé verront leur lien  $\omega_{i,j}$  se renforcer durablement. Ce principe de “coopération” peut s’appuyer sur :

- Une **distance** (ou son inverse),
- Une **similarité** (cosinus, corrélation linéaire, etc.),
- Une **information** ou **co-information** (mutuelle),
- Ou tout autre **critère** de dépendance ou d’utilité mutuelle.

Le **Deep Synergy Learning** n’impose pas un choix rigide, mais définit un **cadre** — le **Synergistic Connection Network** (SCN) — où la mise à jour  $\Delta\omega_{i,j}$  se greffe sur la valeur de  $S(\mathcal{E}_i, \mathcal{E}_j)$ . Les chapitres suivants approfondiront la question du **calcul effectif** de cette synergie, notamment lorsque l’on gère des entités **multimodales** (Chapitre 8) ou de **plusieurs** variables (Chapitre 12). L’idée maîtresse demeure : toute fonction de **coopération** ou d’**interaction** entre deux entités, pourvue d’une signification appropriée au domaine d’application, peut servir de **base** au mécanisme adaptatif du DSL.

#### **2.2.1.3. Relation entre cette Fonction et la Mesure d’Utilité Mutuelle ou d’Interaction**

Après l’exposé des divers exemples (distance, similarité, information mutuelle) pour la **fonction de synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$ , il importe d’analyser la **signification** profonde de ce score dans le **Deep Synergy Learning** (DSL). Cette section [2.2.1.3] aborde la connexion existante entre  $S$  et des concepts plus généraux d’**utilité mutuelle** ou d’**interaction** bénéfique, en formalisant l’idée que la **synergie** reflète une forme de “plus-value” commune aux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ .

##### *A. Notion de Bénéfice Mutuel et d’Utilité*

Considérons un **réseau** adaptatif où l’**interaction**  $(\mathcal{E}_i, \mathcal{E}_j)$  procure un **gain** à la structure globale, par exemple la reconnaissance partagée d’un même cluster ou la contribution à une tâche identique. On peut **formaliser** cette idée au moyen d’une “utilité”  $\text{Util}(\mathcal{E}_i, \mathcal{E}_j)$ , envisagée comme une fonction  $\text{Util}: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^+$ . Il s’agit de quantifier dans quelle mesure la **relation** entre  $i$  et  $j$  se révèle profitable. Dans le DSL, il arrive qu’on assimile

$$S(\mathcal{E}_i, \mathcal{E}_j) = \text{Util}(\mathcal{E}_i, \mathcal{E}_j)$$

ou qu’on emploie une transformation monotone pour s’assurer que  $S \in [0,1]$ . L’essentiel est que plus l’**utilité** perçue est grande, plus la **synergie**  $S$  l’est aussi. Dans un cadre **non supervisé**, la

reconnaissance d'une dépendance mutuelle forte (distance faible, corrélation élevée, information partagée importante) implique un **bénéfice** pour le réseau, lequel renforce alors le lien  $\omega_{i,j}$ .

Exemple : Lorsque la **synergie** consiste en l'**information mutuelle**  $I(X;Y)$ , la “diminution d'incertitude” qu'apporte  $X$  sur  $Y$  s'interprète comme un **gain** (une “utilité”), d'où le renforcement de  $\omega_{i,j}$ . Si la synergie repose sur une **similarité** positive (ex.  $\exp(-\alpha d(\mathbf{x}_i, \mathbf{x}_j))$ ), l'**utilité** réside dans la facilité ou le bénéfice de rassembler deux entités  $\mathbf{x}_i, \mathbf{x}_j$ .

### *B. Lien avec la Coopération et l'Émergence de Clusters*

Dans un **réseau** où les liaisons  $\omega_{i,j}$  se voient renforcées lorsque  $S(\mathcal{E}_i, \mathcal{E}_j)$  est élevée, on retrouve la **logique** de la théorie de la coopération : les unités qui coopèrent s'avèrent plus **connectées** et, par ricochet, le **système** global y gagne. Sur le plan algébrique, des paires  $(i, j)$  pour lesquelles  $S(i, j)$  reste notable finissent par tisser un **sous-réseau** (ou cluster) de pondérations fortes  $\omega_{i,j}$ .

Cette vision recoupe la notion d'**utilité** : regrouper des entités “compatibles” ou “enrichissantes” apparaît comme une stratégie d'amélioration globale (stratégie de “somme positive”). On peut d'ailleurs ajouter une perspective énergétique : considérer la somme (ou l'opposé)

$$\sum_{i < j} \omega_{i,j} S(\mathcal{E}_i, \mathcal{E}_j)$$

en tant que **critère** à maximiser, ce qui amène le réseau à rechercher la configuration  $\{\omega_{i,j}\}$  la plus profitable pour l'ensemble. Les chapitres 4 et 7 présenteront des **théorèmes** expliquant comment, sous certaines hypothèses (symétrie de la synergie, bornes sur  $\eta, \tau$ ), la dynamique converge vers un **minimum local** stable, c.-à-d. une **partition** en clusters où les liens internes sont forts.

### *C. Interaction au-delà de la Simple Corrélation*

Dans certains cas, l'**interaction** entre deux entités outrepassa la corrélation linéaire (Pearson). Un **lien** non linéaire ou conditionnel peut se révéler plus important. De même, l'**information mutuelle** ou la **co-information** peut détecter des dépendances qui échappent à un modèle linéaire. Si  $\mathcal{E}_i$  et  $\mathcal{E}_j$  entretiennent un rapport non trivial (exemple :  $Y = X^2 + \text{bruit}$ ), la corrélation peut être faible, tandis que l'information mutuelle pointe une dépendance claire. Dans un cadre plus vaste encore, si  $\{\mathcal{E}_1, \dots, \mathcal{E}_k\}$  partagent une interaction collective, on peut imaginer une **synergie n-aire** (voir Chapitre 12).

Le **DSL** entend **s'adapter** à de telles configurations. La fonction  $S(\mathcal{E}_i, \mathcal{E}_j)$  doit donc pouvoir accepter une grande variété de mesures d'“utilité” ou de co-dépendance. Cette **flexibilité** fait de ce paradigme un cadre général, capable d'intégrer tantôt une complémentarité sémantique, tantôt une co-occurrence sensorielle, etc.

### *D. Théorème (Existence d'un Score d'Interaction)*

Considérons un ensemble d'entités  $\{\mathcal{E}_i\}$ . Supposons qu'on possède pour chaque paire  $(i, j)$  une mesure d'**interaction**  $\text{Int}(\mathcal{E}_i, \mathcal{E}_j) \geq 0$  reflétant la co-occurrence ou la dépendance. Il est possible de construire une fonction de **synergie**  $S$  grâce à une transformation monotone  $\phi$ . Concrètement, on définit



$$S(\mathcal{E}_i, \mathcal{E}_j) = \phi(\text{Int}(\mathcal{E}_i, \mathcal{E}_j)),$$

où  $\phi$  est une application  $[0, +\infty) \rightarrow [0, 1]$  (ou  $\mathbb{R}^+$ ), strictement croissante et vérifiant  $\phi(0) = 0$ . Ainsi, chaque valeur d'interaction  $\text{Int}(i, j)$  se convertit en un **score**  $S(i, j)$  strictement **non négatif**, aligné sur l'**ordre** (plus  $\text{Int}$  augmente, plus  $S$  augmente). Cette construction prouve qu'**il existe** un moyen d'englober toute forme de mesure d'utilité mutuelle dans la **logique** du DSL, tant que la mesure est monotone et non négative.

## Conclusion

La **fonction de synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  constitue, dans le **DSL**, un **score** traduisant une forme d'**utilité**, de **coopération** ou de **dépendance constructive** entre deux entités. Elle peut relever :

- d'une **distance** (inversée),
- d'une **similarité** (corrélation, cosinus),
- d'une **information mutuelle** plus élaborée,
- ou de tout critère d'**interaction** mutuelle validé par le contexte.

Son rôle consiste à **guider** la dynamique d'auto-organisation du réseau, car seules les paires  $(i, j)$  présentant une synergie élevée verront leurs liaisons  $\omega_{i,j}$  se renforcer. Les chapitres 2.2.2 et 2.2.3 décriront comment la mise à jour de  $\omega_{i,j}(t)$  s'appuie sur  $S(i, j)$  pour donner naissance à un **réseau** (SCN) adaptatif, capable de **former** des **clusters** et de s'auto-organiser autour des entités les plus bénéfiques les unes pour les autres, marquant ainsi l'essence du **Deep Synergy Learning**.

### 2.2.2. Mise à Jour des Pondérations : Formule Générale

Le **Deep Synergy Learning (DSL)** s'appuie sur un réseau d'entités  $\{\mathcal{E}_i\}$  dont les liaisons  $\omega_{i,j}$  évoluent en fonction d'une **fonction de synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$ . La spécificité du DSL réside dans le fait que ces pondérations ne sont pas fixées ni mises à jour par un unique label externe, mais bien par un **processus local** qui adapte la force de chaque lien  $\omega_{i,j}$  selon la valeur de synergie mesurée entre

$\mathcal{E}_i$

et

$\mathcal{E}_j$ .

Dans cette section (2.2.2), on formalise la **règle de mise à jour** la plus courante dans la littérature DSL : une **équation linéaire-additive**, souvent comparée aux règles de Hebb modifiées ou aux dynamiques d'un réseau associatif.

#### 2.2.2.1. Équation $\omega_{ij}(t + 1) = \omega_{ij}(t) + \eta [S(i, j) - \tau \omega_{ij}(t)]$

Dans le cadre du **Deep Synergy Learning (DSL)**, on envisage un **indice de temps** discret  $t = 0, 1, 2, \dots$ , ainsi qu'une matrice de pondérations  $\{\omega_{i,j}(t)\}$ . La mise à jour de ces pondérations, à chaque itération, s'effectue selon la formule

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta[S(i,j) - \tau \omega_{ij}(t)].$$

Les différentes composantes de cette équation se lisent comme suit :

- $\omega_{ij}(t)$  désigne la **force** (ou le poids) de la liaison entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  à l'instant  $t$ .
- $S(i,j)$  désigne la **synergie** (positive ou nulle) entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$ .
- $\eta > 0$  est un **taux d'apprentissage** local, traduisant la rapidité d'ajustement.
- $\tau > 0$  correspond à un **coefficient de décroissance** (ou de dissipation) qui prévient la croissance indéfinie des pondérations.

Cette équation se décompose en deux mécanismes complémentaires :

1. **Renforcement** : la pondération  $\omega_{ij}$  subit un incrément  $\eta S(i,j)$ . Si la synergie  $S(i,j)$  se révèle élevée, la liaison  $\omega_{ij}$  s'accroît.
2. **Décroissance** : le terme  $\eta \tau \omega_{ij}(t)$  vient en déduction, évitant ainsi que les poids  $\omega_{ij}$  ne divergent et ne deviennent arbitrairement grands.

On peut reformuler l'expression en

$$\omega_{ij}(t+1) = (1 - \eta \tau) \omega_{ij}(t) + \eta S(i,j).$$

Cette représentation met en évidence une **combinaison linéaire** de l'ancienne valeur  $\omega_{ij}(t)$ , multipliée par  $(1 - \eta \tau)$ , et d'un **terme constant**  $\eta S(i,j)$ . Pour l'analyse de la convergence, on peut résoudre l'équation stationnaire

$$\omega_{ij}^* = (1 - \eta \tau) \omega_{ij}^* + \eta S(i,j),$$

qui donne

$$\omega_{ij}^* = \frac{\eta}{\eta \tau} S(i,j) = \frac{S(i,j)}{\tau}.$$

Cette solution indique que, si l'on itère la mise à jour et si les valeurs de  $\eta$  et  $\tau$  sont suffisamment réduites ou bien paramétrées,  $\omega_{ij}(t)$  se **stabilise** autour d'un point d'équilibre  $\frac{S(i,j)}{\tau}$ . Dans les chapitres suivants — notamment [Chapitre 4](#) et [Chapitre 7](#) — on examinera plus avant la **convergence** et la façon dont cette dynamique collective (appliquée simultanément à toutes les paires  $(i,j)$ ) peut conduire à la formation de **clusters**.

### Conclusion

La formule ci-dessus illustre de façon limpide le **double mécanisme** propre au DSL :

- L'aspect “**renforcement**” renforce  $\omega_{ij}$  proportionnellement au score  $S(i,j)$ .
- L'aspect “**décroissance**” introduit un frein  $\tau \omega_{ij}(t)$ , évitant l'explosion des liaisons.

La répétition de cette règle itérative, pour  $t = 0, 1, 2, \dots$ , construit progressivement un **réseau** (SCN) au sein duquel les liens  $\omega_{ij}$  **se stabilisent** là où la synergie se maintient, tandis qu'ils se réduisent dans les zones où la synergie demeure faible. Les sections 2.2.2.2 et 2.2.2.3 présenteront les **variantes** (mise à jour multiplicative, inhibition compétitive) ainsi que le **raccord** biologique avec la plasticité neuronale.

### 2.2.2.2. Variantes : Inhibition Compétitive, Mécanisme Multiplicatif vs. Additif

La **formule générale** de mise à jour

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta [S(i,j) - \tau \omega_{ij}(t)] \quad (\text{voir section [2.2.2.1]})$$

illustre le **cas additif**, dans lequel on ajoute un terme  $\eta S(i,j)$  et on retire  $\eta \tau \omega_{ij}(t)$ . Dans la pratique du **Deep Synergy Learning (DSL)**, on rencontre toutefois des **mécanismes de régulation** supplémentaires ou alternatifs, qui modifient la nature de l'adaptation des poids  $\omega_{ij}$ . Deux **familles** de **variantes** se distinguent : l'**inhibition compétitive** et le **mécanisme multiplicatif** (plutôt qu'une simple mise à jour de type linéaire-additif).

#### A. Inhibition Compétitive

La **mise à jour linéaire** du poids  $\omega_{ij}$  présentée en section 2.2.2.1 traite chaque liaison de façon **indépendante**. Or, il est possible d'introduire une **inhibition compétitive**, c'est-à-dire un **effet d'interaction** entre différents liens connectés à la même entité, de sorte que la croissance de certains empêche ou restreigne la croissance d'autres. Dans un réseau biologique, ce phénomène correspond à l'**inhibition latérale** : un neurone fortement actif peut inhiber la plasticité d'autres connexions concurrentes (ou proches).

Dans le **DSL**, l'inhibition compétitive peut prendre la forme de **contraintes** ou de **termes** interagissant avec  $\omega_{ij}$ . Deux approches classiques :

##### 1. Somme limitée :

$$\sum_j \omega_{ij}(t) \leq \Omega_{\max},$$

imposant qu'une entité  $\mathcal{E}_i$  ne puisse maintenir qu'un nombre total (ou une somme totale) de liaisons  $\omega_{i\cdot}$  au-dessus d'un certain seuil.

##### 2. Terme d'inhibition introduit directement dans la règle de mise à jour :

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta \left[ S(i,j) - \tau \omega_{ij}(t) - \sum_{k \neq j} \gamma_{k,j} \omega_{ik}(t) \right],$$

où  $\gamma_{k,j}$  pondère l'**inhibition** exercée par le lien  $\omega_{ik}$  sur la liaison  $\omega_{ij}$ . Ainsi, plus un **nœud**  $\mathcal{E}_i$  a de liaisons puissantes, plus il "freine" la progression de nouvelles connexions.

Cette inhibition compétitive favorise la **sparcification** du réseau et l'apparition de **clusters** plus nets. Même si la synergie  $S(i,j)$  conserve une valeur positive, la liaison  $\omega_{ij}$  peut être contrainte de **décroître** ou de stagner si la "capacité" du nœud  $\mathcal{E}_i$  (ou  $\mathcal{E}_j$ ) est déjà mobilisée par d'autres liens

plus cruciaux. On retrouve ce principe dans divers modèles biologiques et algorithmes inspirés de la neurophysiologie, où il améliore l'**efficacité** de l'apprentissage en évitant un maillage trop complet ou trop diffus du réseau.

### *B. Mécanisme Multiplicatif vs. Additif*

La règle générale

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta[S(i,j) - \tau \omega_{ij}(t)]$$

relève d'une **dynamique additive** : on incrémente  $\omega_{ij}$  d'un terme  $\eta S(i,j)$  et on décrémente d'un terme  $\eta \tau \omega_{ij}(t)$ . Dans certaines approches du **DSL**, on privilégie un **mécanisme multiplicatif** où l'augmentation ou la diminution du poids  $\omega_{ij}$  s'effectue de façon **proportionnelle** à sa valeur courante. Par exemple, on peut adopter une formule telle que

$$\omega_{ij}(t+1) = \omega_{ij}(t) \times (1 - \alpha \tau) + (\text{termes multiplicatifs de } S(i,j)),$$

ou, plus radicalement,

$$\omega_{ij}(t+1) = \omega_{ij}(t) \exp(\alpha[S(i,j) - \tau \omega_{ij}(t)]).$$

Ce style de mise à jour rappelle la **règle de Hebb** : “plus un poids est grand, plus il se renforce lors de co-activation”. Il favorise un **effet exponentiel** : un lien déjà élevé grandit rapidement si la synergie  $S(i,j)$  reste soutenue, tandis qu'un lien faible ne se développe que modérément.

Sur le plan algorithmique, cela peut accélérer la **différenciation** entre liens “dominants” et “inactifs”, donnant lieu à un phénomène “winner-takes-most”. Même si, dans son essence, la logique multiplicative conserve l'idée d'ajuster  $\omega_{ij}(t)$  en fonction de  $S(i,j)$ , elle introduit un **degré** de non-linéarité plus important. La croissance (ou décroissance) dépend non seulement de  $S(i,j)$  mais aussi de la magnitude actuelle de  $\omega_{ij}$ .

### *Conclusion sur les Variantes*

Les règles de mise à jour **diffèrent** de la simple formule linéaire-additive en introduisant soit un terme d'**inhibition compétitive** (permettant une allocation restreinte des ressources du nœud) soit un **mécanisme multiplicatif** (renforçant la proportionnalité de l'apprentissage). Dans un **Synergistic Connection Network (SCN)**, ces variantes :

- **Enrichissent** la dynamique du **DSL** en prévenant la prolifération de liens peu utiles (inhibition) ou en amplifiant la séparation entre liens forts et liens faibles (multiplicatif),
- **Permettent** de mieux contrôler la structure émergente, en assurant la **formation** de clusters nets,
- **Restent** conformes au principe d'**auto-organisation** : la mise à jour demeure un calcul **local** dicté par  $S(i,j)$  et  $\omega_{ij}$ , sans supervision externe.

Les sections 2.2.2.3 et suivantes préciseront l'**interprétation** de ces variations en se référant, par exemple, aux analogies biologiques ou aux conséquences sur la **formation de clusters** dans un réseau DSL.

### 2.2.2.3. Interprétation : Plasticité Locale, Renforcement/Désactivation de Liens

La **mise à jour** des poids  $\omega_{i,j}(t)$  guidée par la **synergie**  $S(i,j)$  peut s'interpréter comme un **mécanisme de plasticité** analogue aux dynamiques neuronales ou aux paradigmes d'auto-organisation. Dans ce cadre, chaque liaison  $\omega_{i,j}$  se **renforce** ou s'**affaiblit** de manière autonome, en fonction de la valeur de synergie mesurée entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ , et sous l'impact d'un terme de décroissance régulateur. Cette section [2.2.2.3] propose une **lecture** (à la fois biologique et algorithmique) de cette dynamique, généralement décrite comme un **processus local** de renforcement/désactivation, fondé sur la co-occurrence ou l'interaction directe entre entités.

#### A. Plasticité Locale

Dans les **réseaux** neuronaux biologiques, la **plasticité** évoque la capacité d'une synapse à ajuster sa "force" en réaction à l'activité simultanée (ou corrélée) de ses neurones pré- et post-synaptiques. L'équation

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta[S(i,j) - \tau \omega_{i,j}(t)]$$

(voir section 2.2.2.1) illustre cette logique : si la **synergie**  $S(i,j)$  est élevée, elle accroît la liaison  $\omega_{i,j}$  d'un terme  $\eta S(i,j)$ . Cette **variation**  $\Delta\omega_{i,j}$  dépend exclusivement de  $\omega_{i,j}(t)$  et de la valeur  $S(i,j)$ , conformément à la philosophie **locale** de la plasticité : il n'y a ni connaissance **globale** ni label supervisé. C'est précisément l'**auto-organisation** : l'ajustement s'opère localement, et la configuration **globale** (éventuelle formation de clusters) émane de l'ensemble de ces modifications éparses. Dans un scénario DSL standard, on découpe le temps en itérations  $t = 0,1,2, \dots$ . Chaque itération applique la règle de plasticité. Dans une version plus biologisante, on passerait en **temps continu** :

$$\frac{d\omega_{i,j}}{dt} = \alpha[S(i,j) - \tau \omega_{i,j}],$$

ce qui revient conceptuellement à la même logique et, en l'absence de perturbations, tend vers l'équilibre  $\omega_{i,j}^* \approx S(i,j)/\tau$ .

#### B. Renforcement vs. Désactivation

Si la quantité  $\tau \omega_{i,j}(t)$  reste **inférieure** à  $S(i,j)$ , alors la mise à jour  $\Delta\omega_{i,j} = \eta[S(i,j) - \tau \omega_{i,j}(t)]$  s'avère **positive** : le poids  $\omega_{i,j}$  **croît**. Cela signifie que, tant que la **synergie**  $S(i,j)$  surpasse un "seuil relatif"  $\tau \omega_{i,j}(t)$ , la liaison se **renforce** pas à pas. De multiples itérations rapprochent  $\omega_{i,j}$  d'un **équilibre**  $\omega_{i,j}^* \approx S(i,j)/\tau$ .

Si, en revanche,  $\tau \omega_{i,j}(t)$  dépasse  $S(i,j)$ , la variation  $\Delta\omega_{i,j}$  devient **négative**. Dans ce cas, la connexion  $\omega_{i,j}$  se **décroît** progressivement, trahissant une **absence** ou un affaiblissement de la

synergie justifiant un grand poids. À la longue,  $\omega_{i,j}$  peut tendre vers une valeur très faible ou proche de zéro, synonyme de désactivation.

Cette dynamique explique pourquoi  $\omega_{i,j}^* = S(i,j)/\tau$  incarne un **point d'équilibre** local : le renforcement et la décroissance s'y compensent exactement.

### *C. Implications pour la Structure Émergente*

Lorsque certains liens  $\omega_{i,j}$  s'avèrent renforcés (synergie  $S(i,j)$  élevée) et que d'autres s'amenuisent (faible synergie, ou trop fortement pénalisés), le **réseau** voit se constituer des **sous-groupes** : on observe des **clusters** (ou micro-clusters) fortement interconnectés. Cela correspond à la formation d'un **état stable** où les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  présentant une synergie notable sont reliées par des pondérations robustes.

Cette approche ne se borne pas à une structure figée, puisque si les **synergies**  $\{S(i,j)\}$  évoluent (par exemple si le système reçoit de nouvelles entités, ou si les co-occurrences se modifient), la **plasticité locale** réactualise  $\omega_{i,j}$ . Le réseau suit donc les **fluctuations** du flux, créant ou dissolvant des connexions selon la tendance du moment (cf. Chapitre 9 sur l'apprentissage continu). Dans la sphère **biologique**, cette logique de co-activation et d'**adaptation** des liaisons renvoie à la **règle de Hebb** ou à d'autres mécanismes (STDP) où les synapses se renforcent si leurs neurones sont fortement corrélés. En IA non supervisée, ces principes rejoignent ceux des **réseaux associatifs**, mais le **DSL** porte l'idée encore plus loin en permettant des fonctions de synergie plus complexes (mesures multimodales, co-information...).

### *Conclusion sur la Plasticité Locale*

La règle

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta[S(i,j) - \tau \omega_{i,j}(t)]$$

peut se lire comme un **processus** de **plasticité** locale, où la synergie  $S$  tient le rôle d'un "signal de co-opération". Les liens présentant un **score**  $S(i,j)$  élevé se **renforcent**, tandis que ceux dépourvus de synergie s'amenuisent ou se désactivent (tendent vers zéro). De fait, un réseau DSL s'oriente vers une **sélection** continue des connexions jugées utiles, maintenant une certaine **résilience** (on peut réactiver un lien si la synergie remonte ultérieurement).

Les chapitres ultérieurs examineront la **convergence** de ces poids et la **stabilisation** d'une structure globale (clusters, attracteurs). Ils aborderont également, dans la section 2.2.2.2, les **variantes** : inhibition compétitive (pour introduire une interaction entre différents liens) et mise à jour multiplicative (pour renforcer l'effet "winner-takes-most"). L'ensemble de ces points ancre la **philosophie** du Deep Synergy Learning : s'appuyer sur une **plasticité locale** pilotée par la synergie pour favoriser l'**émergence** de configurations organisées dans le réseau.

### *2.2.3. Règles de Parsimonie et Seuil de Connexion*

Au fur et à mesure que la **mise à jour** (section 2.2.2) renforce certains liens  $\omega_{i,j}$  selon la synergie  $S(i,j)$ , le **réseau** risque de voir ses pondérations croître un peu partout. Or, dans de nombreux scénarios — qu'il s'agisse de **biologie** (réseaux neuronaux) ou de **DSL** pratique (auto-organisation hétérogène) —, il est souhaitable d'éviter un **graphe trop dense**, qui peut entraîner :

- Une **surcharge** de calcul (trop de liaisons à gérer, complexité accrue),
- Un **manque** de différenciation (le réseau devient quasi complet, il est difficile d'identifier de vrais clusters distincts),
- Des **effets** d'instabilité ou de bruit (des liens résiduels de faible poids saturer inutilement la structure).

La notion de **parsimonie** répond à cela : on ajoute des **règles** (post-traitements ou intégrées à la mise à jour) qui suppriment ou inhibent les liaisons trop faibles, favorisant une structure plus éparse et plus significative.

### 2.2.3.1. Comment Éviter un Graphe Trop Dense : Suppression de Liens sous un Certain $\omega_{\min}$

Dans l'application pratique du **Deep Synergy Learning**, on adopte souvent un **seuil**  $\omega_{\min} > 0$  pour **restreindre** la densité du réseau. Concrètement, après chaque itération ou à des intervalles déterminés, on **supprime** (ou met à 0) les liaisons dont les pondérations  $\omega_{i,j}$  demeurent **inférieures** à  $\omega_{\min}$ . La règle s'écrit :

$$\omega_{i,j}(t+1) = \begin{cases} \omega_{i,j}(t+1), & \text{si } \omega_{i,j}(t+1) \geq \omega_{\min}, \\ 0, & \text{sinon.} \end{cases}$$

Ainsi, les liaisons trop faibles  $\omega_{i,j} < \omega_{\min}$  sont ramenées à zéro, puis (en fonction de la stratégie adoptée) ne sont plus mises à jour ou restent inactives jusqu'à un éventuel regain de **synergie**. En procédant ainsi, on obtient un **réseau** plus **clairsemé**, simplifiant l'identification de **clusters** et la visualisation globale. Sur le plan algorithmique, la démarche se résume à parcourir la matrice  $\omega$  après un certain nombre d'itérations (ou après chaque "epoch") pour **couper** les liaisons trop faibles, ce qui libère les ressources de calcul pour les liens plus importants.

Cette suppression de liens de force négligeable rappelle le **synaptic pruning** dans les réseaux neuronaux biologiques, où des synapses à faible activité finissent par être éliminées, permettant ainsi de recentrer l'efficacité du neurone sur les connexions plus utiles. Dans l'optique d'un **DSL**, un tel mécanisme garantit la **parsimonie** : le réseau se concentre sur les liaisons suffisamment pertinentes, évitant de multiplier les connexions de poids quasi nul. Il existe plusieurs variantes pour implanter ce principe :

- **Seuil unique**  $\omega_{\min}$ .
- **Seuils multiples**  $\omega_{\min 1}, \omega_{\min 2} \dots$ , autorisant un tri par "niveaux" de liens.
- **Pourcentage fixe** : ne conserver que les  $k\%$  de liaisons les plus fortes.

Ces méthodes se montrent particulièrement **bénéfiques** quand le nombre d'entités  $n$  devient grand, car la **matrice**  $\{\omega_{i,j}\}$  peut atteindre une taille  $O(n^2)$ . En réduisant la densité des connexions actives, on **allège** la complexité computationnelle et on **accélère** la convergence de la structure.



### *Conclusion (2.2.3.1)*

Le fait de **supprimer** les liaisons dont  $\omega_{i,j} < \omega_{\min}$  introduit une **parsimonie** salutaire dans le réseau, soutenant une **lisibilité** accrue et un **allègement** des calculs. Les sections 2.2.3.2 et 2.2.3.3 approfondiront les **conséquences** de ce filtrage (amélioration de la clarté des clusters, stabilisation plus rapide) et examineront son **impact** sur la complexité (réduction du nombre de liens à gérer, risque éventuel de négliger certaines synergies modestes mais potentiellement utiles).

### **2.2.3.2. Effets sur la Structure Émergente (Clusters plus Nets, Stabilisation plus Rapide)**

Lorsque la **règle de parsimonie** (section 2.2.3.1) impose de couper ou de désactiver les liaisons  $\omega_{i,j}$  inférieures à un certain seuil  $\omega_{\min}$ , la **topologie** du Synergistic Connection Network (SCN) se réoriente vers un réseau plus **clairsemé**. Ce filtrage sélectif engendre plusieurs **conséquences** majeures sur la **structure** formée et sur la **dynamique** de stabilisation qui en découle.

#### *A. Clusters plus Nets*

En retranchant systématiquement les liaisons  $\omega_{i,j}$  dont la valeur demeure **en deçà** de  $\omega_{\min}$ , on élimine de fait les **ponts** minimes ou les connexions relativement faibles qui ne concourent pas sensiblement à la structure globale. Les liens dont  $\omega_{i,j}$  dépasse  $\omega_{\min}$  deviennent alors les **véritables connexions**, celles qui suggèrent une **synergie** réelle entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ .

Dans une démarche d'**auto-organisation**, on recherche souvent des **clusters** de nœuds fortement interconnectés. La présence d'un seuil  $\omega_{\min}$  clarifie et accentue la séparation entre ces groupements : les liaisons inter-groupes — si elles restent faibles — se trouvent coupées, tandis que les liens intra-groupe, favorisés par une **synergie** non négligeable, franchissent le cap du seuil. Cette **polarisation** entre liens conservés et liens supprimés renforce la visibilité des **composantes** ou des **communautés** dans le SCN.

Par ailleurs, les liaisons de faible intensité pouvaient osciller au cours des itérations, sans réellement influencer sur l'architecture. Les supprimer réduit le risque d'**agrégation** inutile de micro-connexions parasites, et limite par conséquent la formation de signaux aléatoires qui alourdiraient le réseau. De fait, cette **coupe** s'apparente à un **filtrage** : elle fait ressortir les liens les plus significatifs et, par extension, rend la structure plus facilement interprétable.

#### *B. Stabilisation plus Rapide*

Le fait de mettre à zéro une bonne partie des liaisons  $\omega_{i,j}$  jugées trop faibles diminue la **charge** dans la dynamique du réseau. Dès lors, seuls subsistent un nombre restreint de poids **actifs**, sur lesquels se concentrent les mises à jour. En conséquence, la boucle de réajustement se révèle plus **efficace** : il y a moins d'interactions à recalculer, et beaucoup d'anciennes connexions ( désormais à 0 ) ne requièrent plus de suivi continu.

Si, de surcroît, les interactions inter-blocs (ou inter-clusters) étaient de toute façon faibles, elles pouvaient allonger la **convergence** en imposant de petits allers-retours de pondérations. Leur coupure accélère la stabilisation, laissant chaque cluster se structurer **indépendamment**, sans perturbations marginales.



Du point de vue purement algorithmique, si la mise à jour d’une liaison  $\omega_{i,j}$  s’exprime (cf. section 2.2.2.1) par

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta[S(i,j) - \tau \omega_{i,j}(t)],$$

la suppression régulière des liens  $\omega_{i,j}$  restés en dessous de  $\omega_{\min}$  apporte une plus grande **stabilité** : une fois coupées, ces connexions ne fluctuent plus, et les liens restants convergent généralement plus vite vers leur point d’équilibre  $\omega_{i,j}^* \approx S(i,j)/\tau$ . De nombreux travaux empiriques montrent que l’**auto-organisation** se fait plus rapidement quand on élimine régulièrement les connexions trop faibles.

### C. Conclusion sur l’Impact des Règles de Parsimonie

Le simple fait d’imposer un **seuil**  $\omega_{\min}$  (ou toute autre technique de **sparsification**) n’est pas anodin. Il produit des **effets bénéfiques** tant sur la **structure** émergente que sur la **dynamique** de stabilisation :

- Les **clusters** s’affichent de manière **plus nette**, car les liaisons modestes sont coupées et les communautés se démarquent plus facilement.
- La **stabilisation** s’en trouve **accélérée**, puisque l’on maintient un réseau plus léger, évitant l’encombrement par de multiples pondérations moyennes ou faibles.

Ces avantages expliquent pourquoi, dans la pratique du **DSL**, on utilise couramment une procédure de “**thresholding**” (ou “**cut-off**”) pour rendre le **graphe** plus clairsemé et plus interprétable, tout en aidant la convergence. La prochaine section (2.2.3.3) analysera l’**impact sur la complexité** de cette coupe (réduction substantielle du nombre de liaisons à traiter, contrepartie du risque de négliger certaines synergies marginales mais potentiellement utiles).

### 2.2.3.3. Impact sur la Complexité : Gain Potentiel vs. Perte d’Information

Dans le contexte d’un **Synergistic Connection Network (SCN)**, l’application d’une règle de **parsimonie** (voir section 2.2.3.1) conduit à **supprimer** ou à **mettre à zéro** les liaisons  $\omega_{i,j}$  dont la valeur reste inférieure à un certain seuil  $\omega_{\min}$ . Lorsque cette procédure est mise en œuvre, le **réseau** acquiert une structure plus **clairsemée** : nombre de connexions sont annihilées, seules subsistant celles qui dépassent  $\omega_{\min}$ . Cette raréfaction des liens exerce une influence directe sur la **complexité** de l’architecture, ainsi que sur la **qualité** et l’**ampleur** de l’information qui transite dans le SCN. D’une part, la disparition de nombreuses liaisons apporte un **gain** notable en termes de densité et de coûts de calcul ; d’autre part, on risque de se priver de connexions faiblement positives susceptibles de s’avérer utiles. Le présent exposé souligne ces **deux facettes** — la simplification bienvenue et la perte potentielle — et examine les **stratégies** ou **réglages** classiquement envisagés pour concilier **parsimonie** et **préservation** de l’information.

#### A. Gains en Termes de Complexité

Il est fréquent de considérer un ensemble d’entités  $\{\mathcal{E}_1, \dots, \mathcal{E}_n\}$ . Si toutes forment un graphe complet, le nombre de liaisons  $\omega_{i,j}$  à mettre à jour peut tendre vers  $O(n^2)$ . Imposer un **seuil**  $\omega_{\min} >$

0 pour **couper** systématiquement les connexions dont la pondération reste au-dessous de  $\omega_{\min}$  à pour effet :

$$\omega_{i,j}(t+1) = \begin{cases} \omega_{i,j}(t+1), & \text{si } \omega_{i,j}(t+1) \geq \omega_{\min}, \\ 0, & \text{sinon.} \end{cases}$$

Lorsque cette coupure est appliquée, la **densité** du réseau chute. De nombreuses paires  $(i,j)$  reçoivent désormais une pondération nulle et ne participent plus aux calculs, offrant une **réduction** de la complexité globale :

Complexité passera de  $O(n^2)$  à  $O(|\mathcal{A}|)$ ,

où  $\mathcal{A}$  est l'ensemble des **liaisons actives**, c'est-à-dire celles qui franchissent le seuil  $\omega_{\min}$ . Si ce nombre se révèle beaucoup plus modeste que  $n^2$ , la mise à jour se voit grandement **allégée**. Outre le gain algorithmique, ce filtrage favorise la **lisibilité** du réseau, par exemple pour la **détection** et la **visualisation** de clusters, en prévenant la formation d'un graphe trop dense. Il devient ainsi aisé d'identifier les sous-ensembles d'entités réellement interconnectés.

Avantages en un seul paragraphe. L'avantage majeur de cette approche réside dans la diminution drastique des liaisons à gérer, accélérant chaque itération de la mise à jour locale tout en clarifiant la topologie pour l'analyse des clusters. On évite de mobiliser un temps de calcul inutile sur des connexions dont la valeur de pondération reste marginale. Cette simplification s'avère particulièrement pertinente dans des contextes de grande dimension (big data, multimodalité), où la réduction de  $O(n^2)$  à  $O(n)$  ou  $O(n \log n)$  peut faire la différence entre un algorithme inopérant et une solution réalisable en pratique.

### *B. Risque de Perte d'Information*

Si la *parsimonie* via le seuil  $\omega_{\min}$  procure un gain, elle entraîne le **risque** de **couper** des liaisons modérément positives qui auraient pu devenir utiles si on leur laissait le temps de se renforcer. Un lien  $\omega_{i,j} = 0.02$  peut sembler négligeable à l'instant présent, mais il aurait pu se développer ultérieurement, si la **synergie**  $S(i,j)$  augmentait pour des raisons liées à un flux de données changeant. Imposer  $\omega_{i,j} \rightarrow 0$  quand  $\omega_{i,j} < \omega_{\min}$  empêche de ressusciter la liaison à moins qu'on ne prévoise une procédure de "**réactivation**", tolérant le retour d'un lien si  $S(i,j)$  devient subitement plus fort. Sans un tel mécanisme, la décision de couper est souvent définitive ou du moins très contraignante, ce qui peut figer la structure avant qu'une coopération tardive n'ait pu émerger.

Limites en un seul paragraphe. Cette coupure radicale peut aussi segmenter prématurément le réseau en multiples composantes déconnectées, s'il se trouve que la synergie reste faible mais non négligeable entre certains groupes. Un seuil trop ambitieux ( $\omega_{\min}$  trop grand) peut donc conduire à une perte de richesse dans l'exploration des configurations, voire empêcher la coalescence ultérieure de clusters apparentés. Cela constitue la principale contrepartie de la parcimonie, tout en voulant gagner en efficacité, on sacrifie la possibilité de liens modestes mais porteurs d'une information subtile.

### *C. Stratégies de Coupure et Paramétrages*

Il est crucial de trouver un **équilibre** entre l'**exigence** de parsimonie et la **volonté** de préserver les connexions potentielles. Plusieurs méthodes se révèlent utiles :

**Adaptation dynamique du seuil**  $\omega_{\min}$  où l'on commence avec une valeur modeste, puis on l'augmente progressivement selon l'avancement de l'apprentissage. On peut également recourir à :

**Coupes partielles**, plutôt que d'annihiler un lien en dessous de  $\omega_{\min}$ , on le réduit (ex.  $\omega_{i,j} \rightarrow \gamma \omega_{i,j}$ ) afin de laisser une “chance” de relance. Des mécanismes de “réactivation” peuvent se mettre en place si la synergie mesurée  $S(i,j)$  s'élève de nouveau, réinstallant alors la liaison.

La littérature décrit également la possibilité de **conserver** un certain **pourcentage** de liens (5 % ou 10 % des poids maximaux), découpant le reste. Une autre approche consiste à procéder à des coupes plus progressives au fil des époques, ne retenant à chaque étape que les liens les plus élevés.

#### *D. Conclusion sur le Choix du Seuil et la Cohabitation Complexité-Information*

En finalité, l'introduction d'un **seuil**  $\omega_{\min}$  ou d'une méthode équivalente (parsimonie structurelle, maintien d'un top-k) vise à **contenir** la densité du graphe et à **faciliter** la reconnaissance de motifs (clusters, communautés). On profite, en un seul paragraphe, d'une baisse de complexité algorithmique notoire, d'une stabilisation accélérée et d'une clarté accrue dans la cartographie du réseau, au prix d'un danger de couper précocement certaines liaisons prometteuses. Les chapitres d'optimisation et de mise à jour en temps réel (voir [Chapitre 7](#) et [Chapitre 9](#)) aborderont plus en détail les heuristiques adoptées pour calibrer  $\omega_{\min}$ , proposer des palliatifs comme la réactivation de liens et minimiser la “perte” de signaux authentiquement utiles. L'expérience montre que des **ajustements** judicieux, éventuellement adaptatifs, parviennent souvent à concilier les **avantages** (réduction drastique du nombre de liens, stabilisation rapide, meilleure lisibilité) et les **limites** (risque de négliger des connexions latentes ou futures) inhérents à la coupure.

#### **2.2.4. Notions d'États Internes et Auto-Organisation**

Dans l'architecture d'un **Deep Synergy Learning (DSL)**, chaque entité  $\mathcal{E}_i$  est généralement décrite par son ancrage dans un espace  $\mathcal{X}$  (vecteur, symbole, distribution, etc.) et par le poids de ses liaisons  $\{\omega_{i,j}\}$ . Cependant, il est souvent utile d'aller plus loin en autorisant chaque entité à entretenir un **état local** — c'est-à-dire un ensemble de variables ou de “mémoire” qui évolue **en parallèle** de la dynamique des liaisons. Cette idée de “double composante” (poids inter-entités et état interne) contribue à renforcer la capacité d'**auto-organisation**, car elle ajoute un degré de plasticité propre à chaque entité, sans que tout dépende uniquement des pondérations  $\omega_{i,j}$ . La section (2.2.4) détaille comment ce mécanisme peut se mettre en place et quelles conséquences il a sur l'émergence de schémas cognitifs plus riches.

##### **2.2.4.1. Possibilité pour Chaque Entité $\mathcal{E}_i$ de Stocker un État Local, Évoluant en Parallèle**

Il est souvent utile, dans un **Synergistic Connection Network (SCN)**, de ne pas cantonner chaque entité  $\mathcal{E}_i$  à une représentation fixe (par exemple, un simple vecteur  $\mathbf{x}_i$  demeurant invariant). On peut en effet postuler qu'elle possède un **état interne**  $\mathbf{s}_i(t)$  qui se **modifie** au fil des itérations  $t = 0, 1, 2, \dots$ . Ainsi, l'évolution ne concerne plus uniquement les pondérations  $\omega_{i,j}$  (voir la section [2.2.2](#)), mais aussi le **cycle** propre de chaque entité.

## A. Principe Général

On conserve la mise à jour des poids  $\omega_{i,j}(t)$  conformément à la règle (voir section 2.2.2) :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \dots$$

En **parallèle**, chaque entité  $\mathcal{E}_i$  actualise son **état**  $\mathbf{s}_i(t)$  en prenant en compte, par exemple, les informations issues du réseau ou de son activation interne. Il peut exister un ensemble d'équations décrivant la dynamique de l'état local :

$$\mathbf{s}_i(t+1) = \mathbf{F}_i(\mathbf{s}_i(t), \{\omega_{i,j}(t)\}, \{\mathbf{s}_j(t)\}, \dots).$$

La **fonction**  $\mathbf{F}_i$  dépend alors de l'état précédent  $\mathbf{s}_i(t)$ , des pondérations  $\omega_{i,j}(t)$  reliant  $\mathcal{E}_i$  à ses voisins, et éventuellement d'autres signaux extérieurs. Comme dans toute **auto-organisation**, le principe demeure local, chaque entité mettant à jour son état sans qu'un contrôleur global ne dicte de consignes.

Ce postulat accroît sensiblement la **souplesse** du système. Les entités ne sont plus de simples “nœuds passifs” mais deviennent des **agents** disposant d'une **mémoire** ou d'un **contexte** interne. Elles peuvent ainsi **s'adapter** à l'historique, éventuellement conserver des traces de données observées précédemment, moduler la façon dont elles calculent leur synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$  vis-à-vis d'autres entités, ou encore faire varier leur “réceptivité” selon leur état  $\mathbf{s}_i(t)$ . Cela introduit un **couplage** (rétroaction) entre l'état de l'entité et la pondération  $\omega_{i,j}$ .

## B. Richesse des Comportements

Le fait que les entités  $\mathcal{E}_i$  puissent stocker un **état**  $\mathbf{s}_i(t)$  ouvre la voie à des dynamiques plus **complexes**, proches d'un système **multi-agents** ou de **modules cognitifs**. Par exemple, une entité pourrait se comporter en **LSTM local** (voir section 2.2.4.2) ou en bloc mémoire plus rudimentaire. Elle peut assimiler son historique, se doter d'un “contexte” conceptuel, ou enregistrer des **résidus** de perception. Cette organisation “double” (adaptation des poids  $\omega_{i,j}$  d'un côté, adaptation des états  $\mathbf{s}_i(t)$  de l'autre) rappelle certaines structures neuronales biologiques, où la **plasticité** synaptique se combine à l'activité **interne** des neurones (ou groupes de neurones).

Dans le **DSL**, cette évolution conjointe favorise l'émergence de schémas cognitifs distribués. Le réseau ne se limite plus à une **clusterisation** statique. Il peut générer de véritables **boucles** d'activité, des phénomènes de **résonance** ou de **synchronisation**, dans lesquels l'actualisation de  $\mathbf{s}_i(t)$  influence la **synergie**  $S(i, j)$ , laquelle, à son tour, oriente la mise à jour  $\omega_{i,j}(t)$ . Ce double plan adaptatif donne un pouvoir expressif supplémentaire : un nœud  $\mathcal{E}_i$  peut modifier sa **sensibilité** ou son “profil” d'interaction selon ses signaux internes, aboutissant à une **co-organisation** encore plus riche.

On peut écrire : il existe deux niveaux de plasticité qui interagissent,

$$\begin{cases} \omega_{i,j}(t+1) = \omega_{i,j}(t) + \dots, \\ \mathbf{s}_i(t+1) = \mathbf{F}_i(\mathbf{s}_i(t), \{\omega_{i,j}\}, \dots). \end{cases}$$

C'est cette combinaison qui confère au **DSL** son potentiel pour modéliser des fonctionnements plus proches d'une cognition distribuée.

### C. Conclusion sur l'Intérêt d'un État Local

L'incorporation, chez chaque entité  $\mathcal{E}_i$ , d'un **état interne**  $\mathbf{s}_i(t)$  rehausse de beaucoup la **polyvalence** de l'auto-organisation. Les entités ne se réduisent plus à des **identifiants** passifs reliés par des poids : elles deviennent des **agents** disposant d'une **mémoire** propre et d'une **plasticité** interne, suivant un cycle parallèle à celui des poids  $\omega_{i,j}(t)$ . Cette stratégie autorise des comportements dynamiques plus élaborés, la mise en œuvre de **modules cognitifs** (par exemple, “mini-LSTM”), et la création de boucles d'activité auto-organisées (cf. section 2.2.4.2 pour des exemples concrets, et section 2.2.4.3 pour le lien avec l'émergence de schémas cognitifs). En accordant à chaque entité la faculté de faire évoluer son **état**, on confère à l'ensemble du SCN une plus grande **richesse** : les liaisons  $\omega_{i,j}$  ne sont plus seules à se modifier, les **entités** elles-mêmes possèdent un cycle d'apprentissage local. Cette double plasticité (liens + états) accroît la capacité du réseau à se **reconfigurer**, à **mémoriser** des patterns récurrents et à **raisonner** de façon distribuée.

#### 2.2.4.2. Exemple : LSTM Local ou Petite Mémoire Interne

Lorsque l'on propose d'associer à chaque entité  $\mathcal{E}_i$  un **état interne** (se reporter à la section 2.2.4.1), plusieurs voies s'offrent à l'expérimentateur. Certaines approches recourent à un simple **accumulateur** ou à une **variable scalaire**, tandis que d'autres adoptent des dispositifs plus évolués, tels qu'une **cellule LSTM** (Long Short-Term Memory) inspirée des réseaux récurrents profonds. Dans toutes ces configurations, l'objectif demeure de doter chaque entité d'une **capacité à stocker et transformer** les informations de façon autonome, au-delà du seul ajustement des poids  $\omega_{i,j}$ . Les développements ci-dessous illustrent deux options : l'usage d'un **LSTM local** et la création d'une **petite mémoire interne** moins élaborée mais déjà profitable.

##### A. LSTM Local au Sein d'un DSL

Un **LSTM** est une cellule récurrente conçue pour gérer des dépendances de long terme dans les réseaux neuronaux séquentiels. Elle incorpore :

- un **état caché**  $\mathbf{h}(t)$  qui traduit la sortie instantanée de la cellule,
- un **état mémoire**  $\mathbf{c}(t)$  où s'enregistrent des informations plus pérennes,
- plusieurs **portes** (input, forget, output) contrôlant l'apport, l'oubli et l'export de l'information contenue dans  $\mathbf{c}(t)$  et  $\mathbf{h}(t)$ .

Les équations internes de l'LSTM déterminent l'évolution de  $\mathbf{c}(t)$  et  $\mathbf{h}(t)$  à partir d'une entrée  $\mathbf{x}(t)$  et de l'état précédent  $(\mathbf{h}(t-1), \mathbf{c}(t-1))$ .

Dans le **Deep Synergy Learning**, on peut affecter à chaque entité  $\mathcal{E}_i$  sa **cellule LSTM** individuelle :

1. On retient deux **états internes**  $\mathbf{h}_i(t)$  et  $\mathbf{c}_i(t)$ , décrivant respectivement l'état caché et la mémoire à long terme de l'entité  $\mathcal{E}_i$ .
2. À chaque itération,  $\mathcal{E}_i$  reçoit un **signal**  $\mathbf{x}_i(t)$ , par exemple un résumé de l'interaction issue de ses liaisons  $\omega_{i,j}$  et des états  $\mathbf{h}_j(t)$  de ses voisins. Elle procède alors à la mise à jour

$$(\mathbf{h}_i(t), \mathbf{c}_i(t)) = \text{LSTM}(\mathbf{x}_i(t), \mathbf{h}_i(t-1), \mathbf{c}_i(t-1)).$$

3. Le **calcul** de la synergie  $S(i, j)$  peut reposer, en partie, sur  $\mathbf{h}_i(t)$  et  $\mathbf{h}_j(t)$ . On peut, par exemple, définir

$$S(i, j) = \alpha \text{ sim}(\mathbf{h}_i(t), \mathbf{h}_j(t)) + (1 - \alpha) \text{ sim}(\mathbf{x}_i^0, \mathbf{x}_j^0),$$

où  $\mathbf{x}_i^0, \mathbf{x}_j^0$  sont des représentations plus statiques des entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . L'**état LSTM** participe donc à la détermination de la synergie.

4. Les pondérations  $\omega_{i,j}$  évoluent simultanément (conformément au schéma exposé en 2.2.2.1 ou à ses variantes), fournissant un **couplage** entre la dynamique interne des entités et la plasticité des liaisons.

Sur le plan des **retombées**, un LSTM local assure une **rétenion** d'information à plus long terme, permettant à l'entité  $\mathcal{E}_i$  d'**apprendre** et de s'ajuster en se fondant sur un historique étendu. Le réseau gagne en **souplesse**, car une liaison  $\omega_{i,j}$  peut se consolider si les états  $\mathbf{h}_i(t)$  et  $\mathbf{h}_j(t)$  affichent une **similarité** grandissante. L'analogie avec la biologie neuronale (synapses + état interne) s'en trouve renforcée.

### *B. Petite Mémoire Interne : Variante Simplifiée*

Toutes les implémentations du DSL ne requièrent pas la complexité d'un LSTM. Il est tout à fait envisageable d'utiliser une **mémoire** plus rudimentaire, par exemple un **scalaire** ou un **petit vecteur**, pour chaque entité  $\mathcal{E}_i$ . L'idée est de doter  $\mathcal{E}_i$  d'une **capacité** de rappel ou d'accumulation déjà suffisante pour enrichir les comportements auto-organisés.

Si  $\mathbf{s}_i(t) \in \mathbb{R}$  (cas scalaire), on peut l'interpréter comme un accumulateur ou une trace glissante. On peut écrire :

$$\mathbf{s}_i(t+1) = (1 - \beta) \mathbf{s}_i(t) + \beta \cdot \text{input}_i(t),$$

où  $\text{input}_i(t)$  représente les signaux entrants (pondérations voisines, données brutes, etc.). Cette formule confère déjà une **mémoire** à l'entité, mais ne s'encombre pas de la sophistication d'un LSTM.

Dans le cas vectoriel  $\mathbf{s}_i(t) \in \mathbb{R}^d$ , on applique la même logique à chaque composante, autorisant un peu plus de finesse dans la représentation.

Le calcul de la **synergie**  $S(i, j)$  peut alors, en partie, reposer sur  $\text{dist inverse}(\mathbf{s}_i(t), \mathbf{s}_j(t))$  ou un critère analogue, offrant un moyen de tenir compte des **états internes** dans la détermination de la coopération.

Dans bien des cas, cette approche simplifiée suffit pour ajouter un "volet temporel" ou "mémoire" au système, sans la complexité d'un LSTM complet. Toutefois, elle peut se montrer moins flexible (pas de portes d'oubli, etc.) et exiger un réglage manuel plus délicat.

### *C. Intérêt pour l'Auto-Organisation*

Dans un **DSL**, on dispose déjà d'une **plasticité** au niveau des pondérations  $\omega_{i,j}$ . Ajouter une **mémoire** à chaque entité  $\mathcal{E}_i$  institue un second niveau de **plasticité** : d'une part, les liaisons



continuent d'évoluer selon la synergie, et d'autre part, les **entités** gèrent leur état  $\mathbf{s}_i(t)$  en répercutant des observations passées ou des indices internes.

Cette double dynamique (liens + états) se rapproche des idées multi-agents, où chaque “agent” (une entité  $\mathcal{E}_i$ ) possède un **état local** évolutif et des liaisons  $\omega_{i,j}$  en constante régulation. Il en résulte des comportements plus évolués, incorporant une “intelligence” ou un “raisonnement” local distribué. Par exemple, une entité peut retenir dans  $\mathbf{s}_i$  la moyenne récente de signaux perçus, ajuster son calcul de  $S(i,j)$  en conséquence, et influencer la mise à jour  $\omega_{i,j}$ . On voit émerger des phénomènes de synchronisation, de micro-réseaux spécialisés, voire de schémas cognitifs complexes.

#### *D. Conclusion sur l'Intégration d'un État Local*

L'introduction, dans le **Deep Synergy Learning**, d'une mémoire interne (LSTM local ou simple vecteur/scalaire) confère une **dimension supplémentaire** à l'auto-organisation. Les entités ne sont plus de simples points reliés par des poids : elles acquièrent un **cycle** propre, facilitant la **découverte** et la **stabilisation** de structures plus riches. Les liaisons  $\omega_{i,j}$  demeurent ajustées par la synergie, mais cette synergie peut être modulée par le contenu interne  $\mathbf{s}_i(t)$ . On retrouve alors l'analogie d'agents “intelligents” dans un **SCN** dont la plasticité opère tant au niveau des connexions que des entités elles-mêmes. Ainsi, la formation de **clusters** ou de **micro-réseaux** peut s'accompagner d'un raffinement continu (mémoire, attention localisée), rappelant la notion de cognition distribuée. Le DSL gagne donc en expressivité, capable non seulement de regrouper les entités selon leur synergie, mais également de leur laisser la possibilité d'évoluer de l'intérieur, renforçant la richesse de l'**auto-organisation**. Les sections 2.2.4.3 ou les développements ultérieurs décriront plus concrètement ces potentialités, ainsi que les liens avec l'**émergence** de schémas cognitifs plus élaborés.

#### **2.2.4.3. Lien avec l'Émergence de Schémas Cognitifs plus Riches**

Lorsque chaque entité  $\mathcal{E}_i$  se trouve associée à un **état interne** (voir section 2.2.4.1) et que cette **mémoire** ou **dynamique** interne vient compléter la logique de **synergie** décrite en 2.2.4.2, on constate que le **réseau** acquiert la capacité de développer, au fil du temps, des **schémas cognitifs** plus élaborés. Autrement dit, l'**auto-organisation** ne se limite plus à la simple configuration de pondérations  $\omega_{i,j}$ , mais peut susciter des motifs récurrents, des regroupements conceptuels ou des **comportements** distribués analogues à la “cognition distribuée” ou aux “systèmes multi-agents cognitifs”.

Dans un **DSL** classique, on admet déjà que les entités  $\mathcal{E}_i$  ajustent (via leurs représentations ou leurs états) leurs liaisons  $\omega_{i,j}$ , de sorte que celles-ci se **renforcent** si la synergie demeure significative. Dès lors que chaque entité se trouve dotée d'une **composante** mémorielle ou interne, la portée de l'**auto-organisation** s'étend : chaque  $\mathcal{E}_i$  est en mesure de conserver des informations antérieures, au lieu de se cantonner à la valeur instantanée de la synergie. Par ce biais, elle peut moduler son calcul de  $S(i,j)$  ou la manière dont elle relaie ses informations, en fonction d'un **état** interne  $\mathbf{s}_i(t)$ .

Le **réseau** ainsi construit ne se contente plus d'une clusterisation figée : il peut, de fait, développer de **véritables** réseaux de résonances, où chaque entité, grâce à sa “mémoire” ou “conscience” locale, participe à l'émergence de **motifs** globaux plus complexes. D'un point de vue



mathématique, la dynamique ne se restreint plus à l'évolution des pondérations  $\omega_{i,j}(t)$  ; elle englobe également un sous-système  $\{\mathbf{s}_i(t)\}$  dont la synchronisation ou la mise en phase peut conduire à des *routines* ou *schémas* cognitifs. On qualifie souvent un système de “cognitif” quand il gère de l'**information** (que ce soit des perceptions ou des symboles) de manière flexible, qu'il dispose d'une **mémoire** plus ou moins durable et qu'il produit des **décisions** ou **comportements** appuyés sur cet historique interne.

Dans un **DSL** renforcé par des mécanismes de mémoire (section 2.2.4.2), chaque entité  $\mathcal{E}_i$  perçoit la synergie  $\omega_{i,j}$  la reliant à des entités voisines, puis met à jour  $\mathbf{s}_i(t)$  en combinant l'état antérieur et de nouvelles données. Elle propage ensuite ce nouvel état (ou un résumé) dans le **réseau**, influençant la future valeur de  $S(i,j)$ . Ce procédé, généralisé à l'ensemble des entités, assure un **mécanisme distribué** : l'**intelligence** ou la **cognition** ne naît pas d'un composant central, mais de la coordination d'un ensemble de nœuds cognitifs, chacun adaptant en continu sa mémoire et ses liens. Ainsi, on voit émerger des **dynamiques** de synchronisation, l'apparition de “leaders” et “followers”, ou l'organisation spontanée de niveaux auto-gérés au sein d'un unique SCN.

Un *schéma* cognitif peut s'entendre comme un **pattern** récurrent, stable, servant à organiser le traitement de l'information dans le réseau. Par exemple, un schéma perceptif tient la forme d'une règle : “Lorsque je détecte une configuration  $\mathbf{x}$ , j'enclenche la procédure  $\mathbf{y}$ ”. Au sein d'un DSL où chaque entité possède son **état interne**, ce schéma peut s'exprimer via un **état collectif** stabilisé (plusieurs entités conservent des  $\mathbf{s}_i(t)$  cohérents) associé à des liaisons  $\omega_{i,j}$  robustes, c'est-à-dire un motif distribué dans le réseau. Alors que l'auto-organisation non supervisée demeure souvent cantonnée à la formation de **clusters** statiques, la présence d'états internes permet l'apparition de boucles d'activité, de séquences ou de boucles réverbérantes, rappelant la résonance neuronale dans les modèles biologiques. Cela dépasse la simple définition de classes, en autorisant la reproduction de schémas temporels ou la mise en œuvre de “routines cognitives” collectives.

Sur le plan de la **biologie neuronale**, on sait que la formation d'assemblées de neurones (règle de Hebb) peut inclure des états récurrents ou “réverbérants” dans lesquels chaque neurone conserve une certaine **trace** d'activité dépendant de son vécu antérieur. La **cognition** naît de la cohérence entre ces assemblées et de leurs interactions réciproques. Dans les **systèmes multi-agents**, si chaque agent possède un état local évoluant selon un script interne, c'est l'**intelligence collective** qui se développe de leur interaction, parfois vue comme un “protocole” distribué dans le SCN. On peut ainsi observer des phénomènes d'**émergence** de structures conceptuelles, de motifs cognitifs, voire de règles partiellement symboliques, dès lors que ces entités-agents échangent et actualisent leurs mémoires internes.

En combinant la plasticité des poids  $\omega_{i,j}$  et celle de chaque **état**  $\mathbf{s}_i(t)$ , le **DSL** franchit le seuil d'une auto-organisation plus **riche**, où se déploient des *schémas cognitifs* allant au-delà du groupement en clusters. L'hypothèse est qu'un tel réseau “vivant”, dont la mémoire réside à la fois dans les **liaisons** (pondérations) et dans les **états** de chaque nœud, peut développer des patterns de perception, de raisonnement ou d'apprentissage plus **complexes** et plus **adaptatifs**. C'est dans ce sens qu'il offre un **cadre** d'études et d'expérimentations susceptible d'éclairer l'émergence de **comportements** cognitifs distribués, rappelant ou imitant certains principes de la cognition humaine ou animale.

## Conclusion

Lorsque chaque entité  $\mathcal{E}_i$  dispose d'un **état interne**, et que la **synergie** dicte les connexions  $\omega_{i,j}$ , l'**auto-organisation** au sein du **DSL** s'étend vers une dynamique plus "cognitive", génératrice de schémas complexes ou de motifs récurrents. On ne se borne pas à des **clusters** statiques : le réseau peut véhiculer des séquences, des boucles d'activité, des **routines** aux attributs "cognitifs". Ces schémas apparaissent de la **double** plasticité (pondérations  $\omega_{i,j}$  et états internes  $\mathbf{s}_i(t)$ ), illustrant un fonctionnement **distribué**. Les prochains chapitres approfondiront, sous divers angles, la façon dont le **Synergistic Connection Network (SCN)** peut servir de fondement à des comportements apparentés à la **cognition**, nourrissant l'ambition d'un apprentissage non supervisé capable de s'**adapter** et de **raisonner** de manière plus riche.

### 2.2.5. Exemples Illustratifs

Les principes théoriques exposés jusque-là — concernant la définition de la synergie  $S$ , la mise à jour des pondérations  $\omega_{i,j}$ , la parsimonie, ou encore la possibilité d'un état interne pour chaque entité — se concrétisent dans des **exemples** qui permettent de visualiser la **dynamique** du Deep Synergy Learning (DSL) de façon simple et didactique. La section (2.2.5) présente quelques scénarios illustrant ces mécanismes dans des configurations restreintes, afin de mieux comprendre :

- Comment s'applique la **règle de mise à jour** ( $\omega_{ij}(t+1) = \dots$ ) dans un petit réseau,
- L'effet de la **similarité** (ou distance) sur la croissance ou la décroissance des liens,
- L'émergence de **micro-clusters** même dans des systèmes de taille modeste (3 ou 4 entités).

#### 2.2.5.1. SCN Minimal (3 ou 4 Entités) : Démonstration de la Règle de Mise à Jour

Pour **illustrer** de façon simple la dynamique d'un **Synergistic Connection Network (SCN)** dans le cadre d'un **Deep Synergy Learning (DSL)**, on peut se limiter à un **réseau** très restreint de 3 ou 4 entités, par exemple  $\{\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3, \dots\}$ . L'objectif est de **montrer** pas à pas comment les pondérations  $\omega_{i,j}$  évoluent conformément à la **formule** générique expliquée en section 2.2.2 :

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta[S(i,j) - \tau \omega_{ij}(t)].$$

Cette expérience sert de **démonstration** élémentaire : malgré la faible dimension du réseau, elle met en évidence l'esprit du **DSL**, où chaque liaison  $\omega_{i,j}$  se corrige localement en fonction de la synergie  $S(i,j)$ , sans qu'aucune supervision globale n'intervienne.

#### A. Mise en Place d'un Mini-Exemple

Pour clarifier la présentation, on se limite à trois entités  $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3$ . Au temps initial  $t = 0$ , on initialise toutes les pondérations à une petite valeur égale, par exemple :

$$\omega_{12}(0) = \omega_{23}(0) = \omega_{13}(0) = 0.05.$$

On fixe un **taux d'apprentissage**  $\eta = 0.1$  et un **coefficient de décroissance**  $\tau = 0.2$ . On attribue ensuite des **scores** de synergie  $S(i, j)$  purement *ad hoc* :

$$S(1,2) = 0.8, \quad S(1,3) = 0.3, \quad S(2,3) = 0.6.$$

Ici, on suppose  $\mathcal{E}_1$  et  $\mathcal{E}_2$  fortement liées ( $S = 0.8$ ), tandis que  $\mathcal{E}_1$  et  $\mathcal{E}_3$  ne le sont qu'à un niveau modéré ( $S = 0.3$ ). En pratique, ces nombres peuvent provenir d'une **distance**, d'une **similarité** ou d'une **mesure** (cf. section 2.2.1.2), mais on les fixe ici pour la démonstration.

### B. Application de la Règle de Mise à Jour

On effectue la **mise à jour** pour chaque itération  $t = 0, 1, 2, \dots$ . Par exemple, pour la liaison  $\omega_{12}(t)$ , on a :

$$\omega_{12}(t+1) = \omega_{12}(t) + \eta[0.8 - \tau \omega_{12}(t)].$$

De façon analogue, pour  $\omega_{13}$  et  $\omega_{23}$  :

$$\omega_{13}(t+1) = \omega_{13}(t) + \eta[0.3 - \tau \omega_{13}(t)], \quad \omega_{23}(t+1) = \omega_{23}(t) + \eta[0.6 - \tau \omega_{23}(t)].$$

On peut illustrer, par exemple, le premier pas de temps ( $t = 0 \rightarrow 1$ ) pour  $\omega_{12}$ . En notant  $\omega_{12}(0) = 0.05$ , on obtient :

$$\begin{aligned} \omega_{12}(1) &= 0.05 + 0.1[0.8 - 0.2 \times 0.05] = 0.05 + 0.1 \times [0.8 - 0.01] = 0.05 + 0.1 \times 0.79 \\ &= 0.05 + 0.079 = 0.129. \end{aligned}$$

On procédera de même pour  $\omega_{13}(1)$  et  $\omega_{23}(1)$ , puis on itérera à chaque pas de temps ( $t = 1 \rightarrow 2, 2 \rightarrow 3, \dots$ ).

### C. Tendance vers un Équilibre

L'expérience révèle que  $\omega_{12}(t)$  grimpe plus vite (la synergie 0.8 étant la plus élevée) alors que  $\omega_{13}(t)$  stagne ou s'élève doucement (puisque la synergie 0.3 est la plus faible). Lorsqu'on laisse l'itération se dérouler sur plusieurs pas, chaque  $\omega_{i,j}(t)$  se rapproche d'un **point fixe**  $\omega_{i,j}^* \approx S(i, j)/\tau$ , conformément à l'analyse de convergence de la section 2.2.2.1. Ainsi,

$$\omega_{12}^* \approx \frac{0.8}{0.2} = 4.0,$$

même s'il convient de réaliser un nombre significatif d'itérations pour s'en approcher, et de tenir compte du facteur  $\eta$  (taux d'apprentissage) qui peut ralentir ou causer de légères oscillations. L'important est d'observer la **logique** sous-jacente : un score  $S(i, j)$  élevé provoque un renforcement plus prononcé du poids, tant que la décroissance  $\tau \omega_{i,j}(t)$  ne compense pas entièrement le terme  $\eta S(i, j)$ .

### D. Lecture Globale de la Démonstration

Même avec un **mini-exemple** impliquant trois entités  $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3$  et une synergie fixée, on voit clairement comment la **règle** de mise à jour

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta[S(i, j) - \tau \omega_{ij}(t)]$$

entraîne, au fil des itérations, un **rééquilibrage** des pondérations selon les valeurs de  $S(i, j)$ . Les liaisons ayant une synergie plus grande croissent plus vite, celles ayant une synergie modérée stagnent à un niveau bas, et celles potentiellement nulles convergent vers zéro (en l'absence de renfort).

Dans un réseau plus large (4, 5 entités, voire davantage) et avec des définitions de synergie plus sophistiquées (distance euclidienne, similarité cosinus, co-information), le principe demeure identique : chaque connexion  $\omega_{i,j}$  évolue **localement** (cf. sections 2.2.2.2 et 2.2.2.3), et la structure globale (tendance à former des **clusters**) émerge de l'**accumulation** de ces ajustements de bas niveau.

## Conclusion

Le recours à un réseau **très réduit** (trois ou quatre entités) permet d'**exposer** pas à pas la **règle** de mise à jour d'un SCN en **Deep Synergy Learning**, et d'**observer** la façon dont les pondérations  $\omega_{i,j}$  s'orientent vers un **point d'équilibre** proche de  $S(i, j)/\tau$ . Les sections 2.2.5.2 et 2.2.5.3 prolongeront cette démonstration en introduisant des simulations concrètes où la synergie dérive de distances (euclidienne) ou de similarités (cosinus), et en mettant en évidence la **formation** de **micro-clusters** sur des exemples plus larges.

### 2.2.5.2. Courtes Simulations (Distance Euclidienne vs. Similarité Cosinus)

Pour **illustrer** de manière concrète la dynamique d'un **Synergistic Connection Network (SCN)** dans le cadre du **Deep Synergy Learning (DSL)**, il est instructif d'entreprendre de petites **simulations** où la **synergie**  $S(i, j)$  est définie par des formules accessibles (distance euclidienne inversée, similarité cosinus). L'objectif consiste à examiner, sur un **petit ensemble** (par exemple 4, 5 ou 6 vecteurs  $\mathbf{x}_i$ ), la manière dont les pondérations  $\omega_{i,j}$  se renforcent ou diminuent en fonction de la **proximité** vectorielle.

#### A. Contexte de la Simulation

Pour favoriser la **lisibilité**, on peut placer chaque entité  $\mathcal{E}_i$  dans un **plan** à deux dimensions, de sorte que  $\mathbf{x}_i = (x_i, y_i) \in \mathbb{R}^2$ . On peut, par exemple, définir des positions :

$$\mathbf{x}_1 = (1.0, 1.2), \quad \mathbf{x}_2 = (1.3, 0.8), \quad \mathbf{x}_3 = (4.2, 5.1), \quad \mathbf{x}_4 = (4.0, 5.4).$$

On constate que  $\mathbf{x}_1$  et  $\mathbf{x}_2$  se trouvent relativement proches, tandis que  $\mathbf{x}_3$  et  $\mathbf{x}_4$  forment un autre groupe de proximité. Les **pondérations**  $\omega_{i,j}(0)$  sont initialisées à une petite valeur commune (par exemple 0.05). On fixe ensuite :

$$\eta = 0.1, \quad \tau = 0.2,$$

où  $\eta$  est le **taux d'apprentissage** et  $\tau$  le **coefficient de décroissance**.

Afin de **comparer**, on fait tourner la **même** règle de mise à jour,

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta[S(i, j) - \tau \omega_{i,j}(t)],$$

mais en faisant varier  $S(i, j)$  selon deux définitions :

1. **Distance Euclidienne Inversée.** On peut poser

$$S_{\text{dist}}(i, j) = \frac{1}{1 + \|\mathbf{x}_i - \mathbf{x}_j\|}.$$

Ainsi, deux entités plus **proches** (faible distance) reçoivent un score de synergie plus élevé (le dénominateur  $(1 + \text{distance})$  est réduit).

2. **Similarité Cosinus.** On peut employer

$$S_{\text{cos}}(i, j) = \max \left\{ 0, \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\|\mathbf{x}_i\| \|\mathbf{x}_j\|} \right\},$$

mettant l'accent sur l'**angle** entre  $\mathbf{x}_i$  et  $\mathbf{x}_j$ . Deux vecteurs pratiquement parallèles affichent un score élevé, indépendamment de la distance à l'origine.

### B. Déroulement de la Simulation

On effectue généralement un certain nombre (10 ou 20) d'**itérations**. À chaque **step** :

1. On **calcule**  $S(i, j)$  pour chaque paire  $(i, j)$ .
2. On **met à jour**  $\omega_{i,j}(t + 1)$  grâce à la formule ci-dessus.
3. On peut, si on le souhaite, imposer un **seuil**  $\omega_{\min} = 0.01$  (voir section 2.2.3) afin de mettre à zéro les pondérations trop faibles.

Avec la **distance** inversée,  $S_{\text{dist}}(i, j)$  se révèle plus grande si  $\|\mathbf{x}_i - \mathbf{x}_j\|$  est petite. Si, par exemple,  $\mathbf{x}_1$  et  $\mathbf{x}_2$  sont dans un rayon proche, la synergie  $S_{\text{dist}}(1, 2)$  grimpe, et  $\omega_{12}(t)$  s'accroît logiquement. Dans le cas de la **similarité** cosinus  $S_{\text{cos}}(i, j)$ , c'est l'**orientation** qui compte : deux vecteurs quasi parallèles obtiennent une forte corrélation angulaire, et leurs pondérations s'élèvent correspondamment.

Après une dizaine d'**itérations**, la majorité des pondérations  $\omega_{i,j}$  convergent vers un **point** d'équilibre (s'approchant, selon la section 2.2.2.1, de  $S(i, j)/\tau$ ). Les liens jugés peu synergiques (ex. paires avec  $\|\mathbf{x}_i - \mathbf{x}_j\|$  grande ou direction divergente) s'affaiblissent et peuvent tendre vers zéro. L'éventuelle introduction de la **parsimonie** accélère l'obtention d'un **graphe** épars, où n'apparaissent plus que quelques pondérations fortes, signe de la formation de **sous-groupes**.

### C. Analyse et Commentaires

Ces **courtes simulations** révèlent que :

- Lorsque  $S(i, j)$  exploite la **distance** inversée  $1/(1 + \|\mathbf{x}_i - \mathbf{x}_j\|)$ , on favorise la **proximité** géométrique. Des vecteurs rapprochés entraînent une forte synergie et, donc, un renforcement marqué des liens  $\omega_{i,j}$ . On obtient souvent des **regroupements** “spatiaux”.
- Lorsque  $S(i, j)$  recourt à la **similarité** cosinus  $(\max\{0, \mathbf{x}_i \cdot \mathbf{x}_j / (\|\mathbf{x}_i\| \|\mathbf{x}_j\|)\})$ , on regarde l'**angle** entre  $\mathbf{x}_i$  et  $\mathbf{x}_j$ . Deux entités alignées sur une direction analogue reçoivent une synergie élevée, même si elles sont éloignées de l'origine.

Les paramètres  $\eta$  et  $\tau$  influencent la dynamique. Un  $\eta$  trop grand peut générer des oscillations, un  $\eta$  trop petit ralentit la convergence, un  $\tau$  trop grand accélère la décroissance, etc. Dans un contexte de **seuil**  $\omega_{\min}$ , la suppression de liaisons trop faibles clarifie la structure et dévoile rapidement des **sous-groupes** plus nets.

Les simulations “miniatures” montrent la **genèse** de **micro-clusters** : quelques  $\omega_{i,j}$  s’élèvent vite lorsque  $S(i,j)$  est forte, alors que d’autres  $\omega_{i,j}$  tombent à zéro, scellant la partition en petits groupes. On retrouve là la base de l’**auto-organisation** dans le DSL : partir d’un graphe quasi uniforme, puis voir, itération après itération, certains liens se renforcer et d’autres s’éteindre, révélant des **composantes** ou **communautés** cohérentes.

## Conclusion

Même sur un **ensemble** d’entités réduit (4, 5 ou 6 vecteurs), on observe la **dynamique essentielle** du DSL : seules les **paires** dont la synergie  $S(i,j)$  dépasse un certain niveau maintiennent des liaisons élevées, tandis que les liaisons à faible synergie décroissent et s’effacent. Les **micro-clusters** qui en résultent correspondent à des affinités (distance, orientation, etc.). Cette expérimentation rudimentaire fournit une **intuition** claire du fonctionnement d’un **SCN** auto-organisé : on part d’une matrice de pondérations presque uniforme, puis la dynamique  $\omega_{ij}(t+1) = \omega_{ij}(t) + \eta [S(i,j) - \tau \omega_{ij}(t)]$  engendre l’émergence de liens dominants et la formation de **sous-réseaux** stables.

### 2.2.5.3. Premières Observations sur la Formation de Micro-Clusters

Les **courtes simulations** présentées en section 2.2.5.2 – qu’elles s’appuient sur la distance euclidienne inversée ou sur la similarité cosinus – font apparaître qu’après plusieurs itérations de la **règle** de mise à jour des pondérations, un **petit réseau** de 3, 4 ou 5 entités tend à voir naître des **groupes** (ou micro-clusters) fortement connectés. Ce phénomène illustre la **tendance** du Deep Synergy Learning (DSL) à structurer le **Synergistic Connection Network (SCN)** en **clusters**, guidés par des **synergies** relativement élevées.

#### A. Processus d’Émergence des Clusters

Le fait de **définir** une synergie  $S(i,j)$  plus grande pour certaines paires  $(\mathcal{E}_i, \mathcal{E}_j)$  (par exemple en raison de leur proximité ou de leur corrélation statistique) se traduit, dans la **dynamique** décrite en 2.2.2, par une croissance plus rapide des pondérations  $\omega_{i,j}(t)$  correspondantes. Inversement, les paires ne bénéficiant pas d’une synergie élevée voient leurs pondérations décliner ou rester à un niveau bas.

Si, lors des premières itérations, toutes les pondérations  $\omega_{i,j}(0)$  se situent autour d’une même petite valeur, la **règle** de renforcement ( $S(i,j)$ ) et de décroissance ( $\tau \omega_{i,j}$ ) introduit progressivement une “**divergence**” dans l’évolution : quelques liaisons  $\omega_{i,j}$  montent en flèche, d’autres chutent vers zéro. Cet écart de trajectoires amène, dans un réseau **stable**, un **groupement** des entités ayant  $S(i,j)$  plus forte, tandis que les autres connexions s’amointrissent.

Lorsqu’on ne modifie pas la synergie  $S(i,j)$  (c’est-à-dire, si le temps se déroule sans ajout de nouvelles entités ni mise à jour de la mesure de proximité), on constate empiriquement que les pondérations **convergent**. Chaque  $\omega_{i,j}(t)$  se rapproche d’un point fixe, souvent  $\omega_{i,j}^* \approx S(i,j)/\tau$ .



Le **réseau** en fin de phase présente alors quelques couples (ou sous-ensembles) dotés de liens nettement plus forts, mettant en relief leur affinité ou leur synergie.

### *B. Interprétation des Micro-Clusters*

Dans un petit réseau (par exemple 3–5 entités), on peut observer la formation de **micro-clusters**. Un **micro-cluster** est généralement un groupe de 2 ou 3 entités qui se retrouvent reliées par des pondérations élevées, tandis que les connexions vers l’extérieur demeurent faibles ou nulles. Par exemple,  $\{\mathcal{E}_1, \mathcal{E}_2\}$  peut nouer un **couple** au poids  $\omega_{12}$  important, distinct d’un autre binôme  $\{\mathcal{E}_3, \mathcal{E}_4\}$ , affichant ses propres liens robustes.

Ces **sous-ensembles** correspondent à l’idée que  $\mathcal{E}_1$  et  $\mathcal{E}_2$  partagent une **synergie** suffisante (distance réduite, direction vectorielle similaire, etc.). Le réseau “**signe**” cette analogie en **renforçant** les liaisons internes au groupe et en **atténuant** les connexions vers les entités plus éloignées ou moins similaires. Ce mécanisme constitue une **illustration** de la façon dont, à plus grande échelle, le DSL forme spontanément des **clusters**, chaque entité rejoignant la communauté où elle trouve la **synergie** la plus marquée.

Une représentation géométrique du **réseau** dans un plan (positions  $\mathbf{x}_i$  et intensité des liens  $\omega_{i,j}$ ) souligne rapidement ces **groupes** : les liaisons puissantes ressortent, dévoilant la présence de **clusters**. Même sur un petit nombre d’entités, la dynamique de pondérations  $\omega_{i,j}(t)$  montre cette capacité du **DSL** à repérer et à **confirmer** les affinités dominantes, phénomène central de l’**auto-organisation**.

### *C. Prolongements et Évolutions Possibles*

Pour un ensemble plus large (par exemple 10, 20 ou 100 entités), le **même** mécanisme opère à plus grande échelle, créant des **clusters** plus volumineux (voire hiérarchiques). L’adoption d’un **seuil**  $\omega_{\min}$  (voir section 2.2.3) peut accélérer l’obtention d’une **structure** éparse et révéler plus franchement les **groupes**. Sur de très petits réseaux (3 ou 4 entités), l’application d’un seuil conduit à des binômes ou trinômes aux liaisons très intenses, fournissant un exemple minimaliste de **partition**.

Si la synergie  $\{S(i, j)\}$  varie dans le temps (les entités se déplacent, leur mémoire interne change, etc.), la configuration des micro-clusters peut se **réarranger**. Un binôme initialement serré peut se dissoudre, tandis qu’un nouveau couple se consolide. Le DSL, par sa nature locale, suit ces changements en réajustant  $\omega_{i,j}(t)$  selon la nouvelle donne. Même dans un réseau minuscule, on peut ainsi observer des **réorganisations** temporelles et des refontes de clusters en réponse à des variations contextuelles.

Ces **observations** constituent un avant-goût du fonctionnement à échelle plus grande : le **DSL** identifie et **stabilise** des liaisons là où la **synergie** se maintient, dévoilant des **sous-groupes** de haute affinité. Cette logique de “clustering auto-organisé” (ou “micro-clustering” en petit nombre) fonde la démarche du DSL comme algorithme **non supervisé** de regroupement, applicable à des entités diversifiées.

### *Conclusion*

Les simulations à 3–5 entités (section 2.2.5.2) attestent la **capacité** du **Deep Synergy Learning** à **discriminer** les connexions en fonction de la synergie, menant à la formation de **micro-clusters** :



quelques couples ou triplets voient leurs liaisons se hisser vers des valeurs élevées, tandis que d'autres tendent vers zéro. Cette **émergence** de sous-groupes, visible dès les petits réseaux, inaugure la **clusterisation auto-organisée** qu'on observe à plus grande échelle. Elle constitue la base du **DSL** pour l'**apprentissage** non supervisé sur des entités variées (vecteurs, objets symboliques, distributions, etc.).

## 2.3. Hypothèses de Stabilité et Émergence de Clusters

Lorsque l'on considère un **Synergistic Connection Network (SCN)** soumis aux règles de mise à jour décrites (section 2.2), une question naturelle se pose : **le réseau converge-t-il ?** Et si oui, vers quelles formes de topologies ou de structures (clusters, attracteurs) ? La section (2.3) aborde ces problématiques à travers un point de vue “système dynamique” (2.3.1), la possibilité d'attracteurs multiples ou d'oscillations (2.3.2), l'analyse de la formation de clusters (2.3.3) et l'influence du bruit (2.3.4). Nous nous intéresserons également (2.3.5) aux hypothèses fortes et aux limites non résolues.

### 2.3.1. Point Fixe, Attracteurs : Définitions et Existence

L'évolution temporelle du réseau  $\Omega(t)$  — où  $\Omega(t)$  désigne l'ensemble des pondérations  $\{\omega_{i,j}(t)\}$  (et éventuellement d'autres variables d'état) — peut souvent être examinée à travers la **théorie des systèmes dynamiques**. La première sous-section (2.3.1.1) formalise cette perspective ; les suivantes traiteront de l'existence d'un **point fixe** (2.3.1.2) et d'une **analyse locale** de stabilité (2.3.1.3).

#### 2.3.1.1. Approche « système dynamique »

Dans une perspective théorique visant à décrire de manière globale l'évolution des connexions au sein d'un réseau d'entités interconnectées, il est particulièrement instructif d'adopter l'optique d'un système dynamique discret. L'idée centrale consiste à regrouper l'ensemble des pondérations, notées  $\omega_{i,j}$  (éventuellement complétées par des variables représentant des états internes), dans une configuration globale qui sera représentée par le vecteur d'état

$$\Omega(t) = (\omega_{1,2}(t), \omega_{1,3}(t), \dots, \omega_{n-1,n}(t), \dots),$$

où  $t$  désigne l'instant discret (ou l'itération) et où, pour un réseau comportant  $n$  entités, le nombre total de pondérations peut atteindre  $n(n-1)$  (dans le cas orienté) ou  $\frac{n(n-1)}{2}$  (dans le cas non orienté).

L'évolution de l'état du réseau dans le cadre du Deep Synergy Learning (DSL) se traduit par une règle de mise à jour collective, que l'on peut écrire sous la forme d'une équation fonctionnelle :

$$\Omega(t+1) = F(\Omega(t)).$$

Ici, l'opérateur  $F$  représente l'ensemble des règles locales de mise à jour, telles que celles décrites dans la section 2.2.2, où chaque poids évolue selon l'équation

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta[S(i,j) - \tau \omega_{i,j}(t)].$$

La fonction  $S(i,j)$  — qui quantifie la synergie ou l'utilité mutuelle entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  — intervient comme un signal de renforcement, tandis que le terme de décroissance proportionnel à  $\tau \omega_{i,j}(t)$  permet de réguler la croissance des poids. Lorsque l'on regroupe l'ensemble de ces mises

à jour dans un unique opérateur  $F$ , on obtient ainsi une formulation globale de la dynamique du réseau.

#### A. Structure de l'espace d'états

Dans le formalisme usuel de la théorie des systèmes dynamiques discrets, l'évolution se décrit par

$$\mathbf{\Omega}(t + 1) = F(\mathbf{\Omega}(t)), \quad \mathbf{\Omega}(t) \in \mathcal{X} \subseteq \mathbb{R}^D,$$

où  $D$  représente la dimension totale du vecteur d'états (qui, dans notre contexte, est le nombre de pondérations considérées). Plusieurs propriétés de cet espace d'états sont essentielles pour l'analyse :

- **Continuité de  $F$**  : Pour appliquer des résultats classiques tels que le théorème du point fixe de Banach, il est nécessaire que  $F$  soit continue, voire contractante, dans une norme appropriée.
- **Compacité et Bornitude de  $\mathcal{X}$**  : Dans de nombreux modèles, on impose des contraintes (par exemple,  $\omega_{i,j} \geq 0$  ou des saturations maximales) afin que l'ensemble des configurations possibles  $\mathcal{X}$  soit borné. Ces conditions favorisent l'existence d'attracteurs et empêchent la divergence de la norme  $\|\mathbf{\Omega}(t)\|$ .

#### B. Analyse de la dynamique locale et globale

Pour illustrer la dynamique d'un lien individuel, considérons la règle linéaire-additive

$$\omega_{i,j}(t + 1) = (1 - \eta \tau) \omega_{i,j}(t) + \eta S(i, j).$$

Si l'on suppose que les synergies  $S(i, j)$  sont elles-mêmes bornées dans un intervalle  $[0, S_{\max}]$ , il est immédiat de constater que la dynamique unidimensionnelle de chaque  $\omega_{i,j}$  tend vers l'équilibre

$$\omega_{i,j}^* \approx \frac{S(i, j)}{\tau},$$

à condition que les paramètres  $\eta$  et  $\tau$  soient choisis de manière appropriée (c'est-à-dire suffisamment faibles pour garantir une convergence stable et éviter des oscillations indésirables).

Au niveau collectif, l'opérateur  $F$  regroupe l'ensemble de ces équations de mise à jour. Pour étudier la stabilité du système dynamique global, on s'intéresse notamment aux questions suivantes :

4. **Contraction ou Monotonie de  $F$**  : Dans quelle mesure existe-t-il une norme (ou une métrique) pour laquelle l'opérateur  $F$  contracte les distances entre configurations ? Un opérateur contractant garantira, par le théorème du point fixe, l'existence d'un unique point fixe attracteur, vers lequel converge toute trajectoire.
5. **Existence et Stabilité d'un Point Fixe** : Sous quelles conditions sur les paramètres  $\eta$ ,  $\tau$  et la fonction de synergie  $S$  peut-on montrer formellement l'existence d'un équilibre stable  $\mathbf{\Omega}^*$  tel que  $F(\mathbf{\Omega}^*) = \mathbf{\Omega}^*$  ?
6. **Impact des Termes Non Linéaires** : L'introduction de mécanismes supplémentaires, tels que l'inhibition compétitive ou une mise à jour multiplicative, rend la carte  $F$  plus

complexe, car la mise à jour d'un poids peut alors dépendre non seulement de la synergie entre deux entités, mais aussi des interactions avec d'autres connexions (par exemple, via des sommes telles que  $\sum_{k \neq j} \omega_{i,k}$ ). Néanmoins, même dans ce cas, l'analyse se situe dans le cadre général des systèmes dynamiques discrets, et l'on peut rechercher des attracteurs, des cycles ou des comportements plus complexes (éventuellement chaotiques).

### *C. Implications pour l'auto-organisation du SCN*

La formulation  $\mathbf{\Omega}(t+1) = F(\mathbf{\Omega}(t))$  permet d'aborder l'auto-organisation du Synergistic Connection Network (SCN) sous l'angle des systèmes dynamiques. En effet, la convergence des pondérations vers des points fixes ou des attracteurs locaux traduit la formation d'une structure stable au sein du réseau. Ces attracteurs correspondent, dans une interprétation globale, à des configurations dans lesquelles les connexions les plus pertinentes sont renforcées, tandis que les liens moins utiles s'affaiblissent voire disparaissent. Ainsi, l'analyse de la stabilité et de la convergence de  $F$  constitue un outil puissant pour comprendre l'émergence de clusters durables et la robustesse de l'architecture auto-organisée.

Dans des variantes plus avancées du DSL, l'introduction de termes d'inhibition compétitive ou de mécanismes multiplicatifs ne fait qu'enrichir la dynamique, tout en restant dans le cadre itératif général. L'étude des propriétés de l'opérateur  $F$  – par exemple, via la linéarisation autour d'un point fixe et l'analyse de sa jacobienne – permettra d'identifier des conditions précises garantissant l'existence d'un équilibre stable, ou au contraire révélant la présence de cycles limités ou de comportements non linéaires plus complexes.

### *Conclusion*

La vision « système dynamique » du SCN s'exprime par la relation

$$\mathbf{\Omega}(t+1) = F(\mathbf{\Omega}(t)),$$

où  $\mathbf{\Omega}(t)$  regroupe l'ensemble des pondérations (ainsi que, éventuellement, d'autres variables d'état). Cette formulation offre une passerelle vers l'application des outils de la théorie des systèmes dynamiques discrets pour analyser la stabilité, l'existence d'attracteurs et les potentiels phénomènes non linéaires (oscillations, bifurcations, voire chaos) qui pourraient émerger dans le cadre du Deep Synergy Learning. Les développements ultérieurs (voir sections 2.3.1.2 et 2.3.1.3) approfondiront l'analyse formelle de l'existence d'un point fixe  $\mathbf{\Omega}^*$  et de sa stabilité locale, éclairant ainsi la manière dont l'auto-organisation conduit à l'émergence de structures de clusters robustes dans le réseau.

#### **2.3.1.2. Existence d'un point fixe $\mathbf{\Omega}^*$ : conditions théoriques, rôle de $\tau$ et $\eta$**

Dans le cadre de l'approche système dynamique, l'idée centrale consiste à décrire l'évolution globale de la configuration des liaisons d'un réseau en regroupant l'ensemble des pondérations, notées  $\omega_{i,j}(t)$ , dans un vecteur d'état  $\mathbf{\Omega}(t)$ . Ce vecteur, qui peut être considéré comme un élément de l'espace  $\mathcal{X} \subset \mathbb{R}^D$  (où la dimension  $D$  correspond au nombre total de connexions, par exemple  $n(n-1)$  pour un réseau orienté ou  $\frac{n(n-1)}{2}$  dans le cas non orienté), évolue selon une dynamique discrète que l'on peut exprimer par l'équation fonctionnelle

$$\mathbf{\Omega}(t + 1) = F(\mathbf{\Omega}(t)).$$

Ici, l'opérateur  $F$  regroupe de manière globale l'ensemble des règles de mise à jour locales qui, dans le DSL (Deep Synergy Learning), s'expriment typiquement par la relation

$$\omega_{i,j}(t + 1) = \omega_{i,j}(t) + \eta[S(i,j) - \tau \omega_{i,j}(t)],$$

où  $S(i,j)$  représente la synergie ou l'utilité mutuelle entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ ,  $\eta > 0$  est le taux d'apprentissage qui détermine la vitesse d'évolution, et  $\tau > 0$  est un paramètre de décroissance agissant comme régulateur afin d'éviter une croissance indéfinie des poids.

Un point fixe  $\mathbf{\Omega}^*$  de la dynamique est défini par la condition d'invariance

$$\mathbf{\Omega}^* = F(\mathbf{\Omega}^*).$$

La recherche d'un tel point fixe revient à examiner les conditions sous lesquelles la dynamique – issue de la mise à jour locale de chaque connexion – converge vers une configuration stable, c'est-à-dire une configuration pour laquelle les poids ne varient plus au fil des itérations.

#### A. Analyse dans le Cas Additif Simple

Pour mieux illustrer cette idée, considérons d'abord le cas de la mise à jour additive élémentaire. Pour une connexion donnée, la règle de mise à jour s'écrit

$$\omega_{i,j}(t + 1) = (1 - \eta \tau) \omega_{i,j}(t) + \eta S(i,j),$$

en supposant que la valeur  $S(i,j)$  soit constante (ou quasi-stationnaire) dans le temps. Rechercher un équilibre consiste alors à poser  $\omega_{i,j}(t + 1) = \omega_{i,j}(t) = \omega_{i,j}^*$ . La relation d'équilibre devient immédiatement

$$\omega_{i,j}^* = (1 - \eta \tau) \omega_{i,j}^* + \eta S(i,j).$$

En isolant  $\omega_{i,j}^*$  dans cette équation, nous obtenons

$$\eta S(i,j) = \eta \tau \omega_{i,j}^* \quad \Rightarrow \quad \omega_{i,j}^* = \frac{S(i,j)}{\tau}.$$

Ce résultat, qui est immédiatement visible dans le cas unidimensionnel, démontre que chaque poids converge vers un équilibre déterminé par le rapport de la synergie au coefficient de décroissance  $\tau$ . Ainsi, le paramètre  $\tau$  joue un rôle fondamental en modulant l'amplitude finale : si  $\tau$  est élevé, l'équilibre s'établit à des valeurs relativement faibles, tandis qu'un  $\tau$  plus faible permet aux poids d'atteindre des niveaux plus importants. Par ailleurs, le paramètre  $\eta$  conditionne la vitesse de convergence ; un  $\eta$  trop élevé risque de provoquer des oscillations ou des divergences, tandis qu'un  $\eta$  trop faible ralentit le processus de stabilisation.

#### B. Généralisation aux Systèmes Couplés

Dans un réseau complet (ou un Synergistic Connection Network – SCN), la synergie  $S(i,j)$  peut dépendre non seulement de la relation entre les deux entités considérées, mais également des

interactions avec d'autres connexions ou états internes. Dans ce cas, l'opérateur de mise à jour s'écrit de façon plus générale

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i,j; \{\omega_{k,\ell}(t)\}, \{\mathbf{s}_k(t)\}, \dots) - \tau \omega_{i,j}(t)].$$

Le point fixe global  $\mathbf{\Omega}^*$  satisfait alors l'ensemble des équations

$$S(i,j; \{\omega_{k,\ell}^*\}) = \tau \omega_{i,j}^*, \quad \forall (i,j),$$

ce qui conduit à un système non linéaire d'équations. Pour établir l'existence d'un tel point fixe, on peut s'appuyer sur des arguments classiques de théorie des points fixes tels que le théorème de Brouwer ou celui de Schauder. En effet, si l'opérateur  $F$  est continu et que l'ensemble des configurations possibles  $\mathcal{X}$  (souvent défini par des contraintes telles que  $\omega_{i,j} \in [0, \omega_{\max}]$ ) est compact et convexe, alors il existe au moins un point fixe.

L'unicité et la stabilité de ce point fixe dépendent de propriétés supplémentaires de  $F$  – en particulier, si  $F$  est contractant dans une norme appropriée, c'est-à-dire si l'on peut trouver un  $\kappa < 1$  tel que

$$\|F(\mathbf{x}) - F(\mathbf{y})\| \leq \kappa \|\mathbf{x} - \mathbf{y}\|$$

pour tout  $\mathbf{x}, \mathbf{y} \in \mathcal{X}$ . Dans ce cas, non seulement l'existence mais également l'unicité du point fixe est garantie, et toutes les trajectoires convergeront vers celui-ci.

### C. Rôle des Paramètres $\eta$ et $\tau$

Dans la dynamique proposée, le paramètre  $\eta$  agit comme le taux d'apprentissage. Il détermine la rapidité avec laquelle la configuration  $\mathbf{\Omega}(t)$  s'ajuste en réponse au signal de renforcement fourni par  $S(i,j)$  et à l'effet de décroissance proportionnel à  $\tau \omega_{i,j}(t)$ . Un choix judicieux de  $\eta$  est indispensable pour éviter des comportements oscillatoires ou divergents tout en permettant une convergence suffisamment rapide.

Le coefficient  $\tau$  quant à lui intervient comme un régulateur qui « freine » la croissance des poids. En effet, dans la formule additive, il apparaît sous la forme d'un facteur multiplicatif qui, en équilibrant l'influence de la synergie  $S(i,j)$ , fixe l'amplitude finale de chaque pondération par le rapport  $S(i,j)/\tau$ . Une valeur élevée de  $\tau$  force ainsi les poids à converger vers des valeurs plus faibles, ce qui peut être souhaitable pour maintenir un réseau stable et éviter la domination excessive d'un lien sur les autres.

### D. Conclusion

En somme, la formulation du DSL sous la forme d'un système dynamique discret

$$\mathbf{\Omega}(t+1) = F(\mathbf{\Omega}(t))$$

permet d'appréhender la convergence et la stabilisation des liaisons par l'intermédiaire de la recherche d'un point fixe  $\mathbf{\Omega}^*$ . Dans le cas le plus simple d'une mise à jour additive, on démontre que chaque pondération tend vers l'équilibre

$$\omega_{i,j}^* = \frac{S(i,j)}{\tau}.$$

Dans des systèmes plus complexes, où  $S$  dépend de l'ensemble des pondérations et éventuellement d'états internes supplémentaires, l'existence d'un point fixe peut être garantie par les théorèmes classiques de la théorie des points fixes, à condition que l'opérateur  $F$  soit continu et que l'ensemble des états soit compact. Les paramètres  $\eta$  et  $\tau$  jouent alors un rôle déterminant :  $\eta$  régule la vitesse de convergence tandis que  $\tau$  module l'amplitude finale des poids, assurant ainsi que l'équilibre atteint est à la fois stable et cohérent avec la logique de renforcement/décroissance inhérente au DSL.

Cette analyse théorique jette les bases d'une étude approfondie de la stabilité locale – par linéarisation et analyse de la jacobienne de  $F$  – et ouvre la voie à la compréhension des phénomènes d'auto-organisation, tels que la formation de clusters durables au sein du réseau. Les développements ultérieurs s'attachent à préciser les conditions techniques garantissant non seulement l'existence, mais aussi l'unicité et la robustesse de l'attracteur  $\Omega^*$ .

### 2.3.1.3. Analyse locale (linéarisation, Jacobienne) pour évaluer la stabilité

Dans l'étude des systèmes dynamiques discrets appliqués aux réseaux d'auto-organisation du Deep Synergy Learning (DSL), il est essentiel de s'intéresser à la stabilité des points fixes de la dynamique globale. En d'autres termes, une fois qu'une configuration  $\Omega^*$  des poids (et éventuellement d'autres états internes) vérifie

$$\Omega^* = F(\Omega^*),$$

la question cruciale est de déterminer si, en cas de petite perturbation autour de  $\Omega^*$ , le système tend à revenir vers cet équilibre ou s'en éloigne. Pour répondre à cette interrogation, on utilise la technique de linéarisation locale, qui s'appuie sur le calcul de la matrice Jacobienne de l'opérateur  $F$  évaluée en  $\Omega^*$ .

#### A. Linéarisation locale autour d'un point fixe

Soit un système dynamique discret défini par

$$\Omega(t+1) = F(\Omega(t)), \quad \Omega(t) \in \mathbb{R}^D,$$

où  $D$  représente la dimension totale de l'espace d'états, par exemple le nombre total de pondérations  $\omega_{i,j}$  (éventuellement augmenté par d'autres variables telles que les états internes  $\mathbf{s}_i$ ). Un point fixe  $\Omega^*$  est tel que

$$\Omega^* = F(\Omega^*).$$

Pour étudier la stabilité locale de ce point fixe, on introduit une perturbation infinitésimale  $\delta\Omega(t)$  et on écrit

$$\Omega(t) = \Omega^* + \delta\Omega(t).$$

En supposant que  $\delta\Omega(t)$  reste suffisamment petit, il est alors possible de linéariser la fonction  $F$  au voisinage de  $\Omega^*$  en négligeant les termes d'ordre supérieur. On obtient ainsi :



$$\mathbf{\Omega}(t + 1) = F(\mathbf{\Omega}^* + \delta\mathbf{\Omega}(t)) \approx F(\mathbf{\Omega}^*) + DF(\mathbf{\Omega}^*) \delta\mathbf{\Omega}(t),$$

où  $DF(\mathbf{\Omega}^*)$  désigne la matrice Jacobienne (de dimension  $D \times D$ ) de  $F$  évaluée en  $\mathbf{\Omega}^*$ . Puisque  $F(\mathbf{\Omega}^*) = \mathbf{\Omega}^*$ , on peut réécrire :

$$\mathbf{\Omega}(t + 1) \approx \mathbf{\Omega}^* + DF(\mathbf{\Omega}^*) \delta\mathbf{\Omega}(t).$$

En soustrayant  $\mathbf{\Omega}^*$  des deux côtés, la dynamique des perturbations est donnée par :

$$\delta\mathbf{\Omega}(t + 1) = DF(\mathbf{\Omega}^*) \delta\mathbf{\Omega}(t).$$

La stabilité locale du point fixe  $\mathbf{\Omega}^*$  est alors entièrement caractérisée par les valeurs propres de la matrice Jacobienne  $DF(\mathbf{\Omega}^*)$ . En effet, pour un système discret, le point fixe est localement stable (c'est-à-dire, asymptotiquement stable) si et seulement si toutes les valeurs propres  $\lambda$  de  $DF(\mathbf{\Omega}^*)$  satisfont la condition

$$|\lambda| < 1.$$

Si au moins une valeur propre vérifie  $|\lambda| > 1$ , toute petite perturbation dans la direction associée se verra amplifiée, rendant ainsi le point fixe instable. En présence de valeurs propres de module exactement égal à 1, la stabilité locale devient marginale et des comportements plus complexes (cycles, bifurcations) peuvent apparaître.

#### *B. Jacobienne dans le cas “additif” simple*

Pour illustrer concrètement le procédé, considérons le modèle additif simple de mise à jour présenté en section 2.2.2.1 :

$$\omega_{i,j}(t + 1) = \omega_{i,j}(t) + \eta [S(i, j) - \tau \omega_{i,j}(t)],$$

que l'on peut réécrire sous la forme

$$\omega_{i,j}(t + 1) = (1 - \eta \tau) \omega_{i,j}(t) + \eta S(i, j).$$

Si l'on suppose que la fonction de synergie  $S(i, j)$  est constante (ou varie très lentement) et qu'elle ne dépend pas directement des autres poids, la dynamique de chaque connexion se réduit à un système unidimensionnel dont le point fixe satisfait

$$\omega_{i,j}^* = (1 - \eta \tau) \omega_{i,j}^* + \eta S(i, j).$$

Une simple manipulation conduit à

$$\eta S(i, j) = \eta \tau \omega_{i,j}^* \Rightarrow \omega_{i,j}^* = \frac{S(i, j)}{\tau}.$$

La dérivée de la fonction de mise à jour par rapport à  $\omega_{i,j}$  est donnée par

$$\frac{\partial \omega_{i,j}(t + 1)}{\partial \omega_{i,j}(t)} = 1 - \eta \tau.$$

La stabilité de ce point fixe est alors garantie si

$$|1 - \eta \tau| < 1,$$

ce qui se traduit par la condition

$$0 < \eta \tau < 2.$$

Dans un réseau complet, si chaque connexion évolue indépendamment selon la même loi, la matrice Jacobienne  $DF$  sera diagonale, avec  $1 - \eta \tau$  sur chacune de ses entrées diagonales, et ses valeurs propres seront toutes égales à  $1 - \eta \tau$ . Dans le cas où des interactions entre différentes pondérations apparaissent (par exemple, par le biais d'un mécanisme d'inhibition compétitive ou via l'influence des états internes), la Jacobienne ne sera plus strictement diagonale et des termes hors-diagonale, liés aux dérivées partielles  $\frac{\partial S(i,j)}{\partial \omega_{p,q}}$ , viendront modifier sa structure. Toutefois, le critère fondamental reste le même : l'ensemble des valeurs propres doit être inférieur à 1 en module pour garantir la stabilité locale.

### *C. Interprétation : perturbations mineures autour du point fixe*

La relation obtenue

$$\delta \Omega(t + 1) = DF(\Omega^*) \delta \Omega(t)$$

indique que la réponse du système à une perturbation infinitésimale est linéairement déterminée par la Jacobienne évaluée en  $\Omega^*$ . En d'autres termes, si l'ensemble des valeurs propres de  $DF(\Omega^*)$  se trouve dans le disque unité de  $\mathbb{C}$ , alors toute perturbation se contracte au fil du temps et le système revient asymptotiquement à son point fixe. Cette condition, formulée en norme opératoire par

$$\| DF(\Omega^*) \| < 1,$$

est l'exigence fondamentale pour la stabilité locale du point fixe. Dans le cas du modèle additif simple, cette condition se traduit par  $0 < \eta \tau < 2$ . Dans des systèmes plus complexes, le couplage entre les pondérations et l'influence d'autres variables induisent des contributions supplémentaires dans la Jacobienne, mais l'analyse reste structurée par le même principe : étudier la contraction locale des perturbations via les valeurs propres de la matrice dérivée.

### *D. Conclusion sur l'analyse locale*

La linéarisation autour d'un point fixe  $\Omega^*$  permet d'obtenir une description locale de la dynamique par le biais de la matrice Jacobienne  $DF(\Omega^*)$ . Le critère de stabilité locale, essentiel dans la théorie des systèmes dynamiques discrets, se réduit à l'exigence que toutes les valeurs propres  $\lambda$  de cette Jacobienne vérifient

$$|\lambda| < 1.$$

Dans le cas simple où la mise à jour est additive et  $S(i, j)$  est constant, cela se traduit par la condition  $|1 - \eta \tau| < 1$ , soit  $0 < \eta \tau < 2$ . Lorsque le modèle est étendu pour intégrer des dépendances entre les pondérations et/ou l'influence d'états internes, la structure de la Jacobienne se complexifie, mais l'analyse demeure basée sur le contrôle de la norme de  $DF$  autour de  $\Omega^*$ . Cette méthode analytique constitue ainsi un outil puissant pour évaluer la stabilité locale du point fixe et, par

conséquent, pour anticiper la convergence ou la divergence des trajectoires du système en réponse à de petites perturbations.

## Conclusion

En résumé, l'analyse locale par linéarisation – via le calcul de la matrice Jacobienne  $DF(\mathbf{\Omega}^*)$  – fournit un cadre théorique rigoureux pour évaluer la stabilité du point fixe  $\mathbf{\Omega}^*$  dans un système dynamique discret. Le point fixe est localement attractif si et seulement si toutes les valeurs propres de cette Jacobienne sont strictement inférieures à 1 en module. Dans le modèle additif simple, cette condition se traduit par la contrainte  $0 < \eta \tau < 2$ , garantissant ainsi que toute petite perturbation autour de  $\mathbf{\Omega}^*$  sera amortie au fil des itérations. En présence de couplages supplémentaires, bien que la structure de la Jacobienne devienne plus complexe, la méthodologie reste identique : contrôler la contraction locale permet de justifier la stabilité de la configuration d'équilibre.

Ce résultat est fondamental pour le Deep Synergy Learning, car il assure que, sous un choix judicieux des paramètres  $\eta$  et  $\tau$  et sous des hypothèses de bornage et de continuité, la dynamique des poids converge vers des configurations stables, ouvrant la voie à l'émergence de clusters durables et à l'auto-organisation du réseau. Les développements ultérieurs exploreront en profondeur les aspects liés à la multiplicité des attracteurs, aux cycles limités ou encore aux phénomènes de bifurcation qui peuvent survenir lorsque le système est soumis à des rétroactions non linéaires plus complexes.

### 2.3.2. Attracteurs Multiples et Oscillations

L'analyse locale (2.3.1) révèle que, autour d'un **point fixe** donné, on peut évaluer la stabilité via la linéarisation. Toutefois, même si un **point fixe**  $\mathbf{\Omega}^*$  est localement stable, rien ne garantit qu'il soit **unique** : en fonction du **caractère non linéaire** de la mise à jour (couplages, inhibition compétitive, synergies dépendant de plusieurs liaisons), il est fréquent de rencontrer **plusieurs** attracteurs ou même des **dynamismes** cycliques ou chaotiques. La section 2.3.2 s'intéresse à ces phénomènes, en commençant par la possibilité de **minima (ou attracteurs) multiples** (2.3.2.1), avant d'aborder les **cycles et pseudo-chaos** (2.3.2.2), puis les méthodes pour caractériser ces comportements (2.3.2.3).

#### 2.3.2.1. Possibilité de *minima* multiples : le SCN peut converger vers des topologies différentes selon les conditions initiales

Dans l'approche dynamique adoptée pour le Synergistic Connection Network (SCN), la mise à jour itérative des pondérations conduit à une configuration globale représentée par le vecteur  $\mathbf{\Omega}$ . La dynamique globale du réseau est décrite par l'équation

$$\mathbf{\Omega}(t + 1) = F(\mathbf{\Omega}(t)),$$

et un point fixe  $\mathbf{\Omega}^*$  satisfait la condition d'invariance

$$\mathbf{\Omega}^* = F(\mathbf{\Omega}^*).$$

Dans un tel contexte, la nature non linéaire de  $F$  peut engendrer un paysage complexe d'énergie ou de potentiel qui n'est pas convexe. En conséquence, plusieurs solutions à l'équation du point fixe peuvent exister, chacune correspondant à une configuration stable distincte des pondérations.

Une des implications majeures de cette multiplicité des minima est que l'espace des conditions initiales se décompose en différents bassins d'attraction. Autrement dit, selon l'état initial du système, ou même selon l'ordre d'arrivée des données dans un cadre d'apprentissage en flux, le SCN pourra converger vers l'un ou l'autre de ces attracteurs. Ces configurations, qui représentent des topologies de liaisons différentes, traduisent ainsi la possibilité qu'un même réseau, soumis aux mêmes règles de mise à jour, puisse finalement présenter des partitions ou des clusterisations distinctes.

Parmi les facteurs qui contribuent à l'émergence de minima multiples dans un SCN, on peut citer :

- **La structure de la fonction de synergie  $S(\dots)$  :**

Lorsque  $S(i, j)$  dépend de plusieurs variables ou intègre des couplages non linéaires (par exemple, par l'intermédiaire de mécanismes d'inhibition compétitive ou de dépendances  $n$ -aires), le paysage d'énergie associé au réseau présente de multiples vallées. Chaque vallée correspond à une configuration où les poids ont convergé vers des valeurs stables, et ces configurations peuvent être équivalentes en termes de score global.

- **La présence de rétroactions non linéaires :**

Si la règle de mise à jour intègre des termes contextuels – par exemple, une synergie modulée par la somme des poids entrants ou par des états internes des entités – le couplage entre les connexions renforce le caractère non convexe du paysage. Ainsi, plusieurs arrangements cohérents peuvent satisfaire les équations d'équilibre, chacun étant localement stable.

- **Les états internes des entités :**

L'inclusion d'un vecteur d'état interne  $\mathbf{s}_i$  pour chaque entité, qui influence et est influencé par les pondérations  $\omega_{i,j}$ , apporte une dimension supplémentaire à la dynamique. Cette double rétroaction (entre poids et états internes) peut conduire à l'existence de plusieurs régimes d'auto-organisation, c'est-à-dire à des minima multiples correspondant à différentes topologies d'interconnexion.

Pour illustrer ces idées, considérons un exemple schématique. Supposons que nous disposions de quatre entités  $\{\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3, \mathcal{E}_4\}$  et que la structure de la synergie entre elles permette deux configurations stables équivalentes. Dans un premier scénario, les entités  $\mathcal{E}_1$  et  $\mathcal{E}_2$  forment un cluster tandis que  $\mathcal{E}_3$  et  $\mathcal{E}_4$  forment un second cluster. Dans un autre scénario, une partition alternative peut se dessiner, par exemple en regroupant  $\mathcal{E}_1$  avec  $\mathcal{E}_3$  et  $\mathcal{E}_2$  avec  $\mathcal{E}_4$ . Ces deux configurations, bien que différentes, peuvent présenter des scores globaux (ou niveaux d'énergie)

comparables. Le choix de l'attracteur final dépendra alors des conditions initiales et de l'ordre dans lequel les données arrivent, illustrant ainsi la sensibilité du système aux conditions de départ.

Un autre aspect important est que, dans un modèle intégrant des mécanismes d'inhibition compétitive, chaque entité est contrainte de ne maintenir qu'un nombre limité de liaisons fortes. Cette restriction peut favoriser l'émergence de plusieurs arrangements attracteurs, car plusieurs combinaisons de connexions fortes peuvent satisfaire cette contrainte tout en optimisant localement le score de synergie.

En conclusion, la possibilité d'obtenir des minima multiples dans un SCN découle de la nature non convexe et non linéaire de la fonction  $F$  qui gouverne l'évolution du vecteur d'états  $\Omega$ . Les attracteurs multiples traduisent la présence de différents bassins d'attraction dans l'espace des conditions initiales, de sorte que le réseau peut converger vers des topologies distinctes selon l'historique ou l'ordre d'arrivée des données. Cette caractéristique est particulièrement pertinente pour l'auto-organisation et la formation de clusters, car elle souligne que la structure finale du réseau n'est pas unique mais dépend d'un processus d'apprentissage sensible aux conditions de départ et aux rétroactions internes. La section suivante (2.3.2.2) examinera des cas de cycles et de comportements dynamiques complexes, tandis que les analyses ultérieures approfondiront l'impact de ces multiples attracteurs sur la robustesse et la plasticité du SCN.

### 2.3.2.2. Apparition éventuelle de cycles ou de régimes “pseudo-chaotiques” dans certains paramétrages (discussion d'exemples)

Dans le cadre du Deep Synergy Learning (DSL), la dynamique globale du réseau est décrite par la relation itérative

$$\Omega(t + 1) = F(\Omega(t)),$$

où  $\Omega(t)$  représente la configuration complète des pondérations  $\{\omega_{i,j}(t)\}$  (éventuellement complétée par d'autres états internes). Lorsque les règles de mise à jour intègrent des non-linéarités – par exemple, via l'inhibition compétitive, des couplages multiples ou la dépendance aux états internes  $\mathbf{s}_i$  – la carte  $F$  acquiert une complexité telle que l'évolution du système ne converge pas nécessairement vers un point fixe unique. Au lieu de cela, plusieurs comportements dynamiques peuvent émerger, parmi lesquels l'apparition de cycles périodiques ou de régimes « pseudo-chaotiques ».

#### A. Pourquoi un SCN peut-il présenter des cycles ou du pseudo-chaos ?

La clé de l'apparition de telles dynamiques réside dans la richesse non linéaire de la fonction de mise à jour. En effet, si la règle de mise à jour d'un poids  $\omega_{i,j}$  ne se contente pas de répondre de manière linéaire au signal de synergie  $S(i, j)$  et au terme de décroissance proportionnel à  $\tau \omega_{i,j}(t)$ , mais intègre également des contributions contextuelles (par exemple, la somme des autres poids connectés à la même entité ou des rétroactions issues des états internes), alors la fonction  $F$  devient

suffisamment complexe pour permettre l'existence de plusieurs attracteurs ou même de cycles dynamiques.

Concrètement, si la dérivée locale – autrement dit, l'analyse de la Jacobienne de  $F$  au voisinage d'un point fixe – révèle qu'il existe des directions dans lesquelles le module des valeurs propres dépasse 1, le point fixe se déstabilise. Ce phénomène conduit alors, par des bifurcations (par exemple une bifurcation de flip), à l'émergence de cycles de période 2, 3, etc. Dans des systèmes de dimension plus élevée, des enchaînements de bifurcations successives (doublesments de période) peuvent aboutir à des régimes où la trajectoire ne converge plus vers un point fixe ni même vers un cycle simple, mais oscille de manière irrégulière dans l'espace d'états – phénomène que l'on qualifie alors de pseudo-chaotique.

Un mécanisme typique dans ce contexte est celui de l'inhibition compétitive : lorsque le renforcement d'un lien, par exemple  $\omega_{i,j}$ , entraîne l'inhibition des autres liens  $\omega_{i,k}$  pour  $k \neq j$ , le système peut se retrouver dans une situation de va-et-vient, dans laquelle certains liens alternent leur renforcement et leur affaiblissement de manière cyclique.

### *B. Exemples schématiques*

Pour illustrer ces concepts, considérons quelques exemples simplifiés :

- **Cycle de période 2**

Dans un mini-réseau composé de trois entités, il est envisageable que, lors d'une première itération, le lien  $\omega_{1,2}$  soit renforcé au détriment de  $\omega_{1,3}$  en raison d'un mécanisme d'inhibition. Lors de l'itération suivante, la situation s'inverse : l'affaiblissement de  $\omega_{1,2}$  permet à  $\omega_{1,3}$  de se renforcer. Ce va-et-vient induit un cycle de période 2, où le système oscille entre deux configurations distinctes.

- **Cycle de période 3**

Dans un système avec des interactions plus complexes – par exemple, lorsque plusieurs connexions interagissent simultanément et que la dépendance aux états internes est prise en compte – le réseau peut alors adopter un cycle de trois états distincts  $\Omega_1 \rightarrow \Omega_2 \rightarrow \Omega_3 \rightarrow \Omega_1$ . Ce type de comportement est souvent le résultat de rétroactions multiples qui décalent l'équilibre de manière cyclique.

- **Pseudo-chaos dans un système de grande dimension**

Dans un SCN de haute dimension, où le couplage entre de nombreuses pondérations et états internes induit une dynamique non linéaire complexe, il est possible que le système n'atteigne ni un point fixe ni un cycle périodique simple. La trajectoire de  $\Omega(t)$  peut alors parcourir de façon irrégulière une région de l'espace d'états sans jamais se répéter, évoquant ainsi un régime « pseudo-chaotique » caractérisé par une sensibilité accrue aux conditions initiales et une errance permanente dans l'attracteur.

### *C. Discussion mathématique*

D'un point de vue théorique, la stabilité d'un point fixe dans un système dynamique discret est déterminée par les valeurs propres de la Jacobienne  $DF(\Omega^*)$ . Un point fixe est stable si toutes les valeurs propres vérifient  $|\lambda| < 1$ . Cependant, lorsque, dans certaines directions, une ou plusieurs valeurs propres deviennent supérieures à 1 en module, le point fixe se déstabilise et le système peut alors bifurquer vers un cycle ou, avec des enchaînements successifs de telles bifurcations, vers un comportement chaotique.

Le passage par des cycles de période 2, 4, 8, ... (le phénomène classique de doublement de période) est bien documenté dans la littérature sur les systèmes dynamiques non linéaires, notamment dans le modèle logistique. Bien que la structure exacte de  $F$  dans un SCN soit plus complexe, des heuristiques similaires s'appliquent : un taux d'apprentissage  $\eta$  trop élevé ou un couplage (via des mécanismes d'inhibition ou d'autres rétroactions) trop fort peut conduire à l'overshoot et, par conséquent, à l'émergence de cycles. À l'inverse, un ajustement plus modéré de ces paramètres tend à favoriser la convergence vers un point fixe stable.

### *D. Implications pratiques*

D'un point de vue pratique, l'apparition de cycles ou de régimes pseudo-chaotiques dans le SCN présente plusieurs implications :

- **Non-convergence vers un état stable**

Dans un contexte de clustering ou de partitionnement, la présence de cycles signifie que la structure du réseau ne se stabilise pas, ce qui peut compliquer l'extraction d'une partition définitive.

- **Paramétrage délicat**

Afin d'éviter ces comportements oscillatoires indésirables, il est crucial d'ajuster soigneusement les paramètres tels que le taux d'apprentissage  $\eta$ , le coefficient de décroissance  $\tau$  et l'intensité des mécanismes d'inhibition. Des réglages trop agressifs peuvent, en effet, provoquer des bifurcations multiples et conduire à un comportement pseudo-chaotique.

- **Approche cognitive alternative**

Dans certains modèles inspirés de la biologie ou de la cognition, ces régimes oscillatoires peuvent être interprétés comme des mécanismes d'exploration ou d'attention alternante, où le réseau alterne entre plusieurs configurations d'information. Ces dynamiques, loin d'être de simples défauts, pourraient servir à modéliser des processus cognitifs tels que la résonance neuronale ou la fluctuation de l'attention.



### Conclusion (2.3.2.2)

En résumé, dans un SCN dont la règle de mise à jour est fortement non linéaire – notamment par l’intégration de mécanismes d’inhibition compétitive et de couplages complexes – il est tout à fait envisageable que le système n’atteigne pas une convergence vers un point fixe stable. Au lieu de cela, des cycles (de périodes 2, 3, etc.) ou des régimes « pseudo-chaotiques » peuvent apparaître. Cette dynamique, qui résulte du dépassement de certains seuils dans la Jacobienne locale de  $F$  (c’est-à-dire lorsque  $|\lambda| > 1$  dans certaines directions), illustre la richesse des comportements possibles dans des systèmes non linéaires de grande dimension. D’un point de vue pratique, si l’objectif est de stabiliser la structure du réseau en vue d’une partition claire (clustering stable), il est souvent nécessaire d’ajuster les paramètres du système (notamment  $\eta$  et  $\tau$ ) pour éviter de tels régimes oscillatoires. Cependant, ces comportements peuvent aussi être interprétés positivement dans un cadre cognitif, où l’exploration constante des configurations pourrait être associée à des processus d’adaptation ou d’échantillonnage dynamique.

Ainsi, l’apparition éventuelle de cycles ou de régimes pseudo-chaotiques dans un SCN n’est pas simplement un artefact mathématique, mais reflète la complexité inhérente aux interactions non linéaires du système, et offre une richesse dynamique qui peut être exploitée selon les objectifs visés, que ce soit pour stabiliser l’auto-organisation ou pour modéliser des comportements d’exploration et de résonance cognitive.

### 2.3.2.3. Stratégies pour détecter et caractériser ces phénomènes

Dans le cadre de l’étude des systèmes dynamiques non linéaires appliqués au **Synergistic Connection Network (SCN)**, il est primordial de développer des stratégies rigoureuses permettant de détecter et de caractériser les comportements complexes tels que la convergence vers des attracteurs multiples, l’apparition de cycles périodiques ou encore des régimes de type pseudo-chaotique. La dynamique du SCN s’exprime par l’équation itérative

$$\mathbf{\Omega}(t + 1) = F(\mathbf{\Omega}(t)),$$

où le vecteur  $\mathbf{\Omega}(t) \in \mathbb{R}^D$  représente l’ensemble des **pondérations**  $\omega_{i,j}(t)$  et éventuellement d’autres variables d’état, telles que les **états internes**  $\mathbf{s}_i(t)$ . La complexité inhérente à la fonction  $F$  – qui intègre les effets de la **synergie**  $S(i, j)$ , des mécanismes d’inhibition compétitive, et d’éventuelles dépendances croisées entre les pondérations – peut amener le système à ne pas converger vers un point fixe unique, mais à adopter des comportements oscillatoires ou irréguliers.

Une première approche pour détecter ces phénomènes consiste à réaliser des **simulations numériques**. On initialise le SCN avec différentes conditions initiales  $\mathbf{\Omega}(0)$  et on varie les paramètres clés du modèle, tels que le **taux d’apprentissage**  $\eta$  et le **coefficient de décroissance**  $\tau$ . En observant l’évolution des pondérations  $\omega_{i,j}(t)$  au fil du temps, on peut distinguer trois scénarios typiques. Si, après un nombre suffisant d’itérations, les pondérations convergent vers des valeurs constantes, cela indique l’existence d’un point fixe attracteur. À l’inverse, si les valeurs oscillent entre deux configurations ou plus, on déduit la présence d’un cycle de période correspondante, par exemple en constatant que

$$\mathbf{\Omega}(t + p) \approx \mathbf{\Omega}(t)$$

pour une période  $p$  donnée. Enfin, si aucune périodicité claire ne se dégage et que la trajectoire de  $\mathbf{\Omega}(t)$  évolue de manière erratique, l'hypothèse d'un régime pseudo-chaotique est alors plausible.

Pour les systèmes de faible dimension, il est souvent utile d'effectuer une **projection** de la trajectoire dans un espace à deux ou trois dimensions afin de visualiser les trajectoires de  $\mathbf{\Omega}(t)$ . Par exemple, si l'on considère que  $D$  est réduit, on peut représenter graphiquement la suite  $\{\mathbf{\Omega}(t)\}_{t=0}^T$  dans  $\mathbb{R}^3$  pour observer d'éventuelles boucles ou motifs répétés. Dans des réseaux de grande dimension, des outils de visualisation tels que des **cartes de chaleur** ou des **animations** peuvent être employés pour suivre l'évolution des composantes individuelles des vecteurs  $\mathbf{\Omega}(t)$ .

La théorie des systèmes dynamiques offre également des outils formels pour la caractérisation des comportements non convergents. Lorsqu'un point fixe  $\mathbf{\Omega}^*$  est identifié, on peut calculer la **matrice Jacobienne**  $DF(\mathbf{\Omega}^*)$  qui, localement, linéarise la dynamique autour de  $\mathbf{\Omega}^*$  par la relation

$$\delta\mathbf{\Omega}(t+1) = DF(\mathbf{\Omega}^*) \delta\mathbf{\Omega}(t),$$

où  $\delta\mathbf{\Omega}(t) = \mathbf{\Omega}(t) - \mathbf{\Omega}^*$ . La stabilité locale du point fixe est alors déterminée par les valeurs propres  $\lambda$  de  $DF(\mathbf{\Omega}^*)$ . Si l'une des valeurs propres satisfait  $|\lambda| > 1$ , le point fixe se déstabilise dans la direction correspondante, ce qui peut entraîner l'apparition d'un cycle de période 2 (par exemple, via une bifurcation de flip) ou de comportements plus complexes lorsque plusieurs valeurs propres dépassent 1 en module.

Afin de quantifier la présence de régimes chaotiques, on peut recourir au calcul de l'**exposant de Lyapunov maximal**  $\lambda_{\max}$ , défini par

$$\lambda_{\max} = \lim_{t \rightarrow \infty} \frac{1}{t} \ln \frac{\|\delta\mathbf{\Omega}(t)\|}{\|\delta\mathbf{\Omega}(0)\|}.$$

Un exposant de Lyapunov positif indique une sensibilité extrême aux conditions initiales, caractéristique du chaos discret, et confirme ainsi la présence d'un régime pseudo-chaotique.

Pour détecter la **multi-stabilité** du système, il est recommandé de réaliser plusieurs simulations en démarrant à partir de différentes conditions initiales, appelées « multi-run ». Si ces simulations convergent vers des attracteurs différents, cela démontre que le paysage d'énergie du SCN comporte plusieurs minima locaux, chacun disposant de son propre bassin d'attraction. Une autre méthode consiste à perturber légèrement un attracteur supposé stable, par exemple en posant

$$\mathbf{\Omega}(0) = \mathbf{\Omega}^* + \delta\mathbf{\Omega}(0),$$

et en observant si la trajectoire converge toujours vers  $\mathbf{\Omega}^*$  ou si elle bascule vers une autre configuration attractrice.

L'ensemble de ces approches, qu'il s'agisse de simulations numériques, de l'analyse de la matrice Jacobienne ou de l'estimation des exposants de Lyapunov, fournit un cadre complet pour détecter et caractériser les phénomènes de cycles, de multi-attracteurs et de pseudo-chaos dans un SCN. Ces méthodes permettent d'obtenir une compréhension fine des dynamiques non convergentes et, par conséquent, d'adapter le paramétrage du système (notamment les valeurs de  $\eta$  et de  $\tau$ ) afin de favoriser, selon les objectifs, la convergence vers des configurations stables ou, au contraire, de tirer parti des régimes oscillatoires pour modéliser des comportements cognitifs dynamiques.

En somme, la combinaison des techniques de simulation, de linéarisation locale par la matrice Jacobienne et d'analyse quantitative via les exposants de Lyapunov constitue un ensemble d'outils puissants pour détecter et caractériser les phénomènes complexes dans un SCN. Ces stratégies permettent d'identifier si le système converge vers un point fixe attracteur, s'il entre dans un cycle périodique ou s'il adopte un comportement pseudo-chaotique, et offrent ainsi des indications précieuses pour le réglage fin des paramètres du DSL ainsi que pour l'interprétation des processus d'auto-organisation sous-jacents.

### 2.3.3. Analyse de la Formation de Clusters

Dans un **SCN** (Synergistic Connection Network), l'une des manifestations les plus remarquables de la **stabilité** (voire de la multi-stabilité, 2.3.2) est l'**auto-organisation en clusters** plus ou moins durables. Les sections précédentes (2.3.1 et 2.3.2) se concentraient sur la dynamique globale, l'existence de points fixes ou de cycles. Ici (2.3.3), nous nous focalisons plus spécifiquement sur la **notion** de cluster et sur la manière dont ils émergent et se maintiennent. Après avoir défini ce qu'on entend par "cluster stable" (2.3.3.1), nous aborderons (2.3.3.2) les **mesures** (cohésion, modularité) permettant de quantifier la qualité de ces regroupements, puis nous distinguerons (2.3.3.3) les clusters **transitoires** de ceux plus établis (macro-clusters), susceptibles d'apparaître à grande échelle.

#### 2.3.3.1. Concept de "cluster stable" : définition, critères de cohésion interne

Dans le cadre du **Synergistic Connection Network (SCN)**, la notion de **cluster** représente un concept central dans l'étude de l'auto-organisation. Un **cluster**, ou sous-groupe, se définit comme un **ensemble** d'entités  $\mathcal{C} \subseteq \{\mathcal{E}_1, \dots, \mathcal{E}_n\}$  dans lequel les **liens** entre les entités internes, notés  $\omega_{i,j}$  pour  $i, j \in \mathcal{C}$ , sont particulièrement **forts** et traduisent une **synergie** élevée. Parallèlement, les connexions reliant une entité  $\mathcal{E}_i \in \mathcal{C}$  à une entité extérieure  $\mathcal{E}_k \notin \mathcal{C}$  se caractérisent par des valeurs de  $\omega_{i,k}$  nettement plus faibles. Cette dichotomie, intrinsèque à l'auto-organisation, émerge spontanément lorsque, au cours de l'évolution du réseau, certains liens se renforcent tandis que d'autres s'affaiblissent, conduisant à la formation de structures en « blocs » clairement identifiables dans la matrice des pondérations.

La notion de **stabilité** d'un cluster se décline en deux aspects complémentaires qui méritent d'être étudiés en profondeur. D'une part, la **stabilité dynamique** se traduit par le maintien dans le temps des **pondérations internes** du cluster, de sorte que les connexions  $\omega_{i,j}$  entre les entités  $\mathcal{E}_i, \mathcal{E}_j \in \mathcal{C}$  restent élevées et ne se dégradent pas, alors que les liens avec l'extérieur demeurent faibles ou, dans certains cas, disparaissent complètement. Cette propriété dynamique signifie que de petites perturbations, telles que des fluctuations légères de  $\omega_{i,j}$ , n'ont pas la capacité de déstabiliser le sous-groupe ni de modifier substantiellement sa structure. D'autre part, la **cohérence interne** se manifeste par un principe d'auto-renforcement : les entités appartenant à  $\mathcal{C}$  maintiennent une synergie mutuelle qui est systématiquement supérieure à celle qu'elles entretiennent avec des entités extérieures. Cette cohérence peut être formalisée par la définition d'un indice de cohésion, tel que la moyenne des pondérations internes, qui doit rester au-dessus d'un certain seuil pour que l'on puisse parler d'une forte cohésion.

Pour formaliser ces idées, on introduit d'abord l'indicateur de **cohésion minimale interne** défini par

$$\bar{\omega}_{\text{in}}(\mathcal{C}) = \frac{1}{|\mathcal{C}|^2} \sum_{i,j \in \mathcal{C}} \omega_{i,j}.$$

Cet indicateur doit être supérieur à un seuil prédéfini  $\theta_{\text{in}} > 0$  pour garantir que les interactions au sein du cluster sont suffisamment fortes. Parallèlement, l'indicateur de **faible connectivité externe** est défini par

$$\bar{\omega}_{\text{ext}}(\mathcal{C}) = \frac{1}{|\mathcal{C}|(n - |\mathcal{C}|)} \sum_{\substack{i \in \mathcal{C} \\ k \notin \mathcal{C}}} \omega_{i,k},$$

et il est requis que  $\bar{\omega}_{\text{ext}}(\mathcal{C})$  demeure inférieur à un seuil  $\theta_{\text{ext}}$ . La persistance de ces deux critères sur une période temporelle significative, notée par l'intervalle  $t \in [T_0, T_1]$ , assure que le cluster reste stable face aux fluctuations, ce qui indique que l'ensemble  $\mathcal{C}$  agit comme un **attracteur local** dans l'espace dynamique du réseau.

Cette approche permet de traduire de manière formelle le concept de **cluster stable**. On peut exprimer cette stabilité par l'inégalité

$$\frac{1}{|\mathcal{C}|^2} \sum_{i,j \in \mathcal{C}} \omega_{i,j}(t) \gg \frac{1}{|\mathcal{C}|(n - |\mathcal{C}|)} \sum_{\substack{i \in \mathcal{C} \\ k \notin \mathcal{C}}} \omega_{i,k}(t),$$

qui doit être vérifiée de manière persistante pour  $t \in [T_0, T_1]$ . L'interprétation mathématique de cette inégalité consiste à affirmer que la **moyenne interne** des pondérations est bien supérieure à la **moyenne externe**, garantissant ainsi une **séparation nette** entre les interactions intra-cluster et inter-cluster.

Les avantages de cette approche résident dans sa capacité à fournir une mesure quantitative de la cohésion interne et de la séparation externe, ce qui est essentiel pour identifier et analyser les regroupements dans le cadre d'un apprentissage non supervisé. L'utilisation de telles mesures permet non seulement de détecter des clusters stables, mais également de caractériser leur robustesse face aux perturbations. Toutefois, un inconvénient potentiel est que les seuils  $\theta_{\text{in}}$  et  $\theta_{\text{ext}}$  doivent être choisis en fonction de l'échelle typique des pondérations dans le SCN, ce qui peut rendre l'approche sensible à la normalisation des données et aux paramètres du système.

En conclusion, le **concept de “cluster stable”** dans le SCN repose sur la persistance d'un sous-ensemble d'entités  $\mathcal{C}$  dans lequel les **liens internes** sont significativement plus forts que les **liaisons externes**, traduisant ainsi une **stabilité dynamique** et une **cohérence interne** durable. La condition

$$\frac{1}{|\mathcal{C}|^2} \sum_{i,j \in \mathcal{C}} \omega_{i,j}(t) \gg \frac{1}{|\mathcal{C}|(n - |\mathcal{C}|)} \sum_{\substack{i \in \mathcal{C} \\ k \notin \mathcal{C}}} \omega_{i,k}(t)$$

remplit un rôle essentiel pour quantifier cette stabilité et sert de fondement à l'analyse de la **clusterisation** dans un **Deep Synergy Learning**. Les approches développées ici seront enrichies dans les sections ultérieures, notamment dans **2.3.3.2** où des mesures plus élaborées, telles que la **modularité** et d'autres indices de cohésion, seront introduites afin de différencier les clusters transitoires des véritables **macro-clusters** stables dans le réseau.

### 2.3.3.2. Mesures possibles (cohésion, modularité) pour juger la qualité ou la force d'un regroupement

Dans le cadre de l'analyse des **clusters** issus du Synergistic Connection Network (SCN), il est fondamental de disposer de critères quantitatifs permettant de mesurer la **cohésion interne** et la **séparation externe** d'un sous-ensemble d'entités  $\mathcal{C} \subseteq \{\mathcal{E}_1, \dots, \mathcal{E}_n\}$ . Ces mesures visent à déterminer dans quelle mesure les **liaisons**  $\omega_{i,j}$  établies entre les entités au sein du cluster sont significativement plus fortes que celles reliant les membres de  $\mathcal{C}$  aux entités extérieures. Un premier indicateur est la **densité interne** qui évalue la moyenne (ou la somme) des pondérations entre toutes les paires d'entités appartenant à  $\mathcal{C}$ . Par exemple, on définit la densité interne par

$$\bar{\omega}_{\text{in}}(\mathcal{C}) = \frac{1}{|\mathcal{C}|(|\mathcal{C}| - 1)} \sum_{\substack{i,j \in \mathcal{C} \\ i \neq j}} \omega_{i,j}.$$

Cette moyenne quantifie le niveau de **force** des interactions à l'intérieur du cluster et, plus sa valeur est élevée, plus la cohésion interne est importante. En parallèle, il est pertinent de mesurer la **connectivité externe** du cluster en évaluant la moyenne des pondérations entre les entités à l'intérieur de  $\mathcal{C}$  et celles situées dans le complémentaire  $V \setminus \mathcal{C}$  du réseau. On définit ainsi

$$\bar{\omega}_{\text{ext}}(\mathcal{C}) = \frac{1}{|\mathcal{C}|(n - |\mathcal{C}|)} \sum_{\substack{i \in \mathcal{C} \\ k \notin \mathcal{C}}} \omega_{i,k}.$$

L'interprétation de ces deux mesures permet d'établir un critère de séparation, qui peut être exprimé par le **ratio** suivant

$$\text{ratio}(\mathcal{C}) = \frac{\bar{\omega}_{\text{in}}(\mathcal{C})}{\bar{\omega}_{\text{ext}}(\mathcal{C}) + \epsilon},$$

où  $\epsilon > 0$  est une constante très faible introduite pour éviter toute division par zéro. Un ratio élevé indique que le cluster est fortement cohésif par rapport à ses connexions extérieures, ce qui constitue un indice quantitatif fort de **stabilité** et de **qualité** du regroupement.

Une approche complémentaire consiste à recourir à la notion de **modularité**, empruntée à l'analyse des communautés dans les graphes pondérés. Dans ce contexte, la modularité permet de comparer la somme des **liaisons internes** à ce que l'on obtiendrait dans un modèle aléatoire conservant la même distribution des degrés. Pour un graphe pondéré  $G = (V, E)$ , la modularité, dans sa forme Newman-Girvan adaptée, se définit par

$$Q = \frac{1}{2W} \sum_{\substack{i,j \in V \\ \text{même cluster}}} \left[ \omega_{ij} - \frac{k_i k_j}{2W} \right],$$

où  $2W = \sum_{i,j} \omega_{ij}$  représente le total des pondérations du graphe et  $k_i = \sum_j \omega_{ij}$  désigne le degré pondéré du nœud  $i$ . Un score de modularité élevé, souvent supérieur à 0.3 ou 0.4 selon l'échelle, témoigne d'une division du graphe en communautés (ou clusters) qui exhibent des **liaisons internes**

significativement supérieures à ce qui serait attendu par hasard. Cette mesure, en intégrant des références statistiques au modèle aléatoire, offre un cadre rigoureux pour évaluer la **qualité** des regroupements et, par extension, leur pertinence dans l'analyse des données.

Par ailleurs, il existe d'autres indices issus de l'analyse des graphes, tels que la **conductance** ou la **densité relative**, qui permettent d'évaluer la proportion des liaisons sortantes par rapport à l'ensemble des liaisons d'un cluster. La conductance, par exemple, est faible lorsque le cluster présente une forte cohésion interne et une faible connectivité externe. Bien que ces indices soient souvent utilisés en complément de la modularité, leur emploi dans le contexte du SCN permet de disposer d'une palette d'outils pour apprécier de manière fine la **robustesse** d'un regroupement.

Du point de vue pratique, ces mesures jouent un rôle crucial dans le cadre du **Deep Synergy Learning**. En effet, elles servent non seulement à identifier les clusters qui émergent de manière auto-organisée, mais également à comparer différentes partitions obtenues dans des simulations multi-run, notamment lorsque le réseau présente plusieurs attracteurs possibles. Ces outils facilitent ainsi la visualisation et l'analyse des structures en blocs qui se dégagent dans la matrice des pondérations, permettant de regrouper les entités en “macro-nœuds” pour simplifier l'interprétation globale du réseau.

En conclusion, la quantification de la **cohésion interne** et de la **séparation externe** via des mesures telles que  $\bar{\omega}_{\text{in}}(\mathcal{C})$ ,  $\bar{\omega}_{\text{ext}}(\mathcal{C})$  et le **ratio** associé, complétée par le calcul de la **modularité**,

$$Q = \frac{1}{2W} \sum_{\substack{i,j \in V \\ \text{même cluster}}} \left[ \omega_{ij} - \frac{k_i k_j}{2W} \right],$$

fournit un cadre analytique robuste pour juger de la **qualité** et de la **force** d'un regroupement dans un SCN. Ces indicateurs permettent de distinguer les clusters **stables** – qui se caractérisent par une cohésion interne élevée et une séparation nette par rapport au reste du réseau – des configurations transitoires, apportant ainsi des bases solides pour l'analyse des communautés dans le contexte du Deep Synergy Learning. La section suivante (2.3.3.3) approfondira ces concepts en distinguant les clusters transitoires des véritables macro-clusters qui émergent sur le long terme.

### 2.3.3.3. Distinction entre clusters transitoires et macro-clusters à large échelle

Dans le cadre du **Synergistic Connection Network (SCN)**, l'analyse des regroupements d'entités conduit à distinguer deux phénomènes d'auto-organisation qui, bien que fondamentalement liés, diffèrent par leur durabilité et leur rôle dans la structure globale du réseau. Le premier phénomène correspond aux **clusters transitoires**, qui se caractérisent par une **cohésion interne** temporaire et une apparition éphémère dans la dynamique du SCN, tandis que le second se réfère aux **macro-clusters** qui, par leur stabilité et leur persistance sur de longues périodes, se révèlent comme des attracteurs dominants dans l'espace des configurations.

La notion de **cluster transitoire** désigne un sous-ensemble  $\mathcal{C} \subseteq \{\mathcal{E}_1, \dots, \mathcal{E}_n\}$  dont les liaisons internes, représentées par l'ensemble des pondérations  $\{\omega_{i,j} \mid i, j \in \mathcal{C}\}$ , atteignent un niveau élevé pendant un intervalle de temps limité. Formellement, si l'on définit la cohésion interne moyenne par

$$\bar{\omega}_{\text{in}}(\mathcal{C}, t) = \frac{1}{|\mathcal{C}|(|\mathcal{C}| - 1)} \sum_{\substack{i, j \in \mathcal{C} \\ i \neq j}} \omega_{i, j}(t),$$

et la connectivité externe moyenne par

$$\bar{\omega}_{\text{ext}}(\mathcal{C}, t) = \frac{1}{|\mathcal{C}|(n - |\mathcal{C}|)} \sum_{\substack{i \in \mathcal{C} \\ k \notin \mathcal{C}}} \omega_{i, k}(t),$$

alors un cluster transitoire se manifeste par une condition telle que, pour un intervalle temporel restreint  $t \in [T_0, T_1]$ ,

$$\bar{\omega}_{\text{in}}(\mathcal{C}, t) \gg \bar{\omega}_{\text{ext}}(\mathcal{C}, t).$$

Ce regroupement peut apparaître en raison de fluctuations temporaires dans les **états internes** des entités ou lors de phases de transition où de petits sous-groupes se forment brièvement avant de fusionner ou de se dissoudre. Ces phénomènes, bien que révélateurs d'une forte synergie momentanée, ne garantissent pas la pérennité de la structure car de nouvelles mises à jour, l'effet d'une inhibition compétitive ou l'évolution des synergies, peuvent entraîner la disparition rapide de ces regroupements.

À l'inverse, le concept de **macro-cluster** se réfère à un regroupement stable et persistant. Un macro-cluster est défini comme un sous-ensemble  $\mathcal{C}$  dont la cohésion interne, mesurée par

$$\bar{\omega}_{\text{in}}(\mathcal{C}, t),$$

se maintient à un niveau élevé pendant une période prolongée, et dont la connectivité externe,  $\bar{\omega}_{\text{ext}}(\mathcal{C}, t)$ , reste faible de façon durable. On peut caractériser la stabilité d'un macro-cluster par l'inégalité

$$\frac{1}{|\mathcal{C}|^2} \sum_{i, j \in \mathcal{C}} \omega_{i, j}(t) \gg \frac{1}{|\mathcal{C}|(n - |\mathcal{C}|)} \sum_{\substack{i \in \mathcal{C} \\ k \notin \mathcal{C}}} \omega_{i, k}(t),$$

qui doit être vérifiée de manière continue sur un intervalle temporel  $[T_0, T_1]$  suffisamment long pour certifier la **persistance** et la **robustesse** du regroupement. La stabilité d'un macro-cluster s'exprime également par une faible **variance** temporelle de  $\bar{\omega}_{\text{in}}(\mathcal{C}, t)$ , signe que les synergies internes ne fluctuent pas de manière excessive, et par la capacité du cluster à résister à des **perturbations** mineures (par exemple, un léger bruit ajouté aux pondérations).

Pour distinguer concrètement les clusters transitoires des macro-clusters, plusieurs approches méthodologiques peuvent être mises en œuvre. L'utilisation d'une **fenêtre glissante** permet de segmenter l'évolution temporelle en intervalles  $[t, t + \Delta]$  sur lesquels les mesures  $\bar{\omega}_{\text{in}}(\mathcal{C}, t)$  et  $\bar{\omega}_{\text{ext}}(\mathcal{C}, t)$  sont calculées. Si, au sein d'une seule fenêtre, la cohésion interne atteint un pic mais retombe ensuite, on identifie un regroupement transitoire. Par ailleurs, l'application d'un **test de perturbation**, consistant à injecter un bruit faible dans la configuration des pondérations, peut révéler la robustesse d'un macro-cluster : un cluster stable se reconstitue rapidement après perturbation, tandis qu'un regroupement éphémère s'effondre.



Un exemple schématique permet d'illustrer cette distinction. Considérons un SCN de cinq entités où, lors des premières itérations, un petit sous-groupe tel que  $\{\mathcal{E}_1, \mathcal{E}_2\}$  présente une forte cohésion interne, indiquée par un pic temporaire de  $\bar{\omega}_{\text{in}}(\{\mathcal{E}_1, \mathcal{E}_2\}, t)$ . Toutefois, si, lors des itérations suivantes, cette cohésion ne persiste pas et que ces entités finissent par être intégrées dans un regroupement plus vaste, alors  $\{\mathcal{E}_1, \mathcal{E}_2\}$  correspond à un **cluster transitoire**. En revanche, si une partition plus large, par exemple  $\mathcal{C} = \{\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3, \mathcal{E}_4\}$ , se maintient avec un  $\bar{\omega}_{\text{in}}(\mathcal{C}, t)$  élevé et une  $\bar{\omega}_{\text{ext}}(\mathcal{C}, t)$  faible sur une durée prolongée, alors ce regroupement constitue un **macro-cluster** stable.

Les **avantages** d'une telle approche résident dans la possibilité de quantifier de manière rigoureuse la qualité d'un regroupement via des indices mathématiques tels que la densité interne et la **modularité**. Cette modularité, définie par

$$Q = \frac{1}{2W} \sum_{\substack{i,j \in V \\ \text{même cluster}}} \left[ \omega_{ij} - \frac{k_i k_j}{2W} \right],$$

avec  $2W = \sum_{i,j} \omega_{ij}$  et  $k_i = \sum_j \omega_{ij}$ , offre une mesure comparative entre le regroupement observé et ce que l'on pourrait obtenir dans un modèle aléatoire. Un score de modularité élevé indique que les liens internes sont bien supérieurs aux attentes d'un réseau aléatoire, attestant ainsi de la robustesse et de la pertinence du regroupement. Les **limites** de cette méthode résident principalement dans la nécessité d'ajuster les seuils  $\theta_{\text{in}}$  et  $\theta_{\text{ext}}$  en fonction de l'échelle des pondérations dans le SCN, ainsi que dans la sensibilité potentielle des mesures à des fluctuations temporelles qui, dans des systèmes très dynamiques, pourraient rendre la distinction entre clusters transitoires et macro-clusters moins nette.

En conclusion, la distinction entre **clusters transitoires** et **macro-clusters à large échelle** repose sur l'analyse de la persistance et de la robustesse des liens internes par rapport aux connexions externes. Un cluster transitoire est caractérisé par une **cohésion interne** momentanée qui ne se maintient pas sur la durée, tandis qu'un macro-cluster se distingue par une **stabilité dynamique** persistante et une séparation nette des interactions internes et externes. Cette distinction, formalisée par l'inégalité

$$\frac{1}{|\mathcal{C}|^2} \sum_{i,j \in \mathcal{C}} \omega_{i,j}(t) \gg \frac{1}{|\mathcal{C}|(n - |\mathcal{C}|)} \sum_{\substack{i \in \mathcal{C} \\ k \notin \mathcal{C}}} \omega_{i,k}(t),$$

et complétée par des indicateurs tels que la modularité  $Q$ , constitue une base rigoureuse pour l'identification et l'analyse des regroupements au sein d'un SCN. Ces mesures quantitatives permettent ainsi de distinguer les regroupements éphémères, susceptibles de se dissoudre rapidement, des structures stables qui émergent en tant qu'attracteurs durables dans la dynamique d'auto-organisation du réseau. La section suivante (2.3.3.3) approfondira cette distinction en examinant des cas pratiques et en proposant des méthodes complémentaires pour évaluer la qualité des clusters dans un contexte d'apprentissage non supervisé.

### 2.3.4. Influence du Bruit et des Perturbations

L'analyse de la **stabilité** et de la **formation de clusters** (sections 2.3.1 à 2.3.3) s'est déroulée dans un cadre relativement "idéal". Dans la **pratique**, un **SCN** (Synergistic Connection Network) peut faire face à diverses **perturbations** ou à un **bruit** injecté — que ce bruit affecte la **mesure de synergie** (données incertaines, capteurs bruyants, etc.) ou qu'il concerne directement la **dynamique** des pondérations (ajout d'aléas dans la mise à jour). La section 2.3.4 examine comment ces facteurs perturbateurs influent sur la trajectoire du SCN et la robustesse des **clusters** qu'il forme. Après une introduction (2.3.4.1) sur la façon dont le SCN réagit au bruit, on abordera la **résilience** (2.3.4.2) et l'idée qu'un **certain** niveau de bruit peut même \emph{favoriser} l'exploration (2.3.4.3).

#### 2.3.4.1. Comment le SCN réagit à un bruit injecté sur les entités ou sur la fonction de synergie

Dans le cadre du **Synergistic Connection Network (SCN)**, la présence de **bruit** joue un rôle crucial dans la dynamique d'auto-organisation. La perturbation aléatoire, qui peut être introduite à divers niveaux — que ce soit dans la **représentation** des entités  $\mathbf{x}_i$  ou de leurs **états internes**  $\mathbf{s}_i$ , dans le calcul de la **fonction de synergie**  $S(i, j)$  ou encore directement dans la **mise à jour** des **pondérations**  $\omega_{i,j}$  — influence directement la capacité du réseau à converger vers une configuration stable. En effet, le SCN évolue selon l'équation de mise à jour additive

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i, j) - \tau \omega_{i,j}(t)],$$

où le terme  $\eta [S(i, j) - \tau \omega_{i,j}(t)]$  détermine l'ajustement des poids. L'injection d'un **bruit** peut modifier l'issue de cette évolution de plusieurs manières.

Lorsqu'un **bruit** est injecté dans la **représentation** des entités, par exemple en ajoutant une perturbation aléatoire à chaque vecteur  $\mathbf{x}_i$ , la **distance** ou la **similarité** utilisée pour calculer  $S(i, j)$  sera modifiée. Autrement dit, même si la structure intrinsèque des entités demeure identique, les mesures telles que

$$S(i, j) = \exp(-\alpha \|\mathbf{x}_i - \mathbf{x}_j\|)$$

peuvent varier en raison d'une fluctuation aléatoire dans  $\mathbf{x}_i$  ou  $\mathbf{x}_j$ . Ainsi, deux entités qui étaient auparavant considérées comme proches peuvent temporairement apparaître éloignées, ce qui affecte négativement la **consolidation** de leurs liens et peut retarder, voire empêcher, la formation d'un regroupement stable.

De même, si le **bruit** est directement appliqué sur la **fonction de synergie**, on peut modéliser cette perturbation par l'introduction d'un terme aléatoire  $\varepsilon_{i,j}(t)$  dans le calcul de la synergie. On écrira alors

$$S_{\text{obs}}(i, j) = S_{\text{réel}}(i, j) + \varepsilon_{i,j}(t),$$

ce qui implique que la valeur utilisée pour mettre à jour les poids est bruitée. Même lorsque les représentations  $\mathbf{x}_i$  et  $\mathbf{x}_j$  sont parfaitement stables, la présence de  $\varepsilon_{i,j}(t)$  conduit à une fluctuation dans le signal de renforcement, induisant ainsi des variations aléatoires dans l'évolution de  $\omega_{i,j}$ .

Il est également possible d'injecter le **bruit** directement dans la **mise à jour** des pondérations. Dans ce cas, l'équation de mise à jour se modifie en ajoutant un terme perturbateur  $\xi_{i,j}(t)$  :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i,j) - \tau \omega_{i,j}(t)] + \xi_{i,j}(t).$$

Même si la valeur de  $S(i,j)$  reste exacte, ce terme  $\xi_{i,j}(t)$  génère des fluctuations supplémentaires qui font osciller les poids autour de leur point d'équilibre théorique  $\omega_{i,j}^* \approx S(i,j)/\tau$ .

Les **effets** du **bruit** dépendent de son amplitude relative au terme de correction  $\eta [S(i,j) - \tau \omega_{i,j}(t)]$ . Lorsque le **bruit** est modeste, la dynamique fondamentale de renforcement et de décroissance domine, de sorte que le réseau parvient à stabiliser ses pondérations malgré de légères fluctuations, voire à explorer de nouveaux minima locaux dans l'espace d'état, ce qui peut parfois éviter de rester bloqué dans une configuration sous-optimale. En revanche, si le **bruit** atteint une amplitude comparable, voire supérieure, à celle du terme de correction, la convergence vers un point fixe est perturbée. Le système adopte alors une dynamique où les pondérations évoluent de façon erratique, ce qui se traduit par un comportement de **diffusion aléatoire** ou par un « mouvement brownien » autour du point d'équilibre.

Les **avantages** potentiels d'un bruit modéré résident dans sa capacité à permettre au SCN de s'extraire de minima locaux, favorisant ainsi l'exploration de configurations alternatives qui pourraient être plus adaptées. Toutefois, les **limites** de cette approche apparaissent lorsque le bruit devient trop important, car il perturbe significativement la stabilité du réseau et empêche la formation de **clusters** robustes, rendant ainsi l'ensemble de la structure instable et difficilement interprétable.

En conclusion, l'effet du **bruit** dans un SCN est à la fois ambivalent et paramétrable. Si l'injection d'un bruit faible dans les **entités**, dans la **fonction de synergie** ou dans la **mise à jour** des **pondérations** permet de renforcer l'exploration du paysage d'énergie et d'éviter des blocages, une perturbation trop forte empêche la stabilisation des liens et retarde, voire annule, la formation de structures cohérentes. Les paramètres  $\eta$  et  $\tau$  jouent ici un rôle déterminant en comparant leur contribution au signal de correction avec l'amplitude du bruit, déterminant ainsi si le **Deep Synergy Learning** parvient à conserver une topologie cohérente ou s'il bascule dans un régime plus erratique.

#### 2.3.4.2. Résilience vs. Sensibilité : Cas d'Exemple

Dans l'analyse des **Synergistic Connection Networks (SCN)**, l'étude de la réponse du système à l'injection de bruit permet d'appréhender la robustesse de sa dynamique d'auto-organisation. On s'intéresse particulièrement à la manière dont le système réagit lorsque des perturbations aléatoires sont introduites dans le calcul de la **fonction de synergie**  $S(i,j)$  ou directement dans l'équation de mise à jour des **pondérations**  $\omega_{i,j}$ . Pour illustrer ces phénomènes, considérons d'abord la règle de mise à jour dite « additive » qui constitue le cœur de la dynamique du SCN :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i,j) - \tau \omega_{i,j}(t)],$$

où  $\eta > 0$  représente le taux d'apprentissage et  $\tau > 0$  le coefficient de décroissance. Dans le cas idéal, et en l'absence de perturbations, cette mise à jour conduit à une convergence locale des pondérations vers un **point fixe** théorique défini par

$$\omega_{i,j}^* \approx \frac{S(i,j)}{\tau}.$$

Pour modéliser l'impact des perturbations, on introduit un terme de bruit  $\xi_{i,j}(t)$  dans la dynamique, ce qui donne l'équation suivante :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i,j) - \tau \omega_{i,j}(t)] + \xi_{i,j}(t).$$

Le terme  $\xi_{i,j}(t)$  représente des fluctuations aléatoires qui viennent perturber la trajectoire de convergence de  $\omega_{i,j}(t)$ . La comparaison entre l'amplitude du terme de correction,  $\eta [S(i,j) - \tau \omega_{i,j}(t)]$ , et celle du bruit  $\xi_{i,j}(t)$  détermine si le système affiche une **résilience** ou une **sensibilité** accentuée face aux perturbations.

#### Exemple de Résilience :

Supposons un mini-SCN composé de quatre entités, notées  $\{\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3, \mathcal{E}_4\}$ , pour lesquelles les synergies internes  $S(1,2)$  et  $S(3,4)$  sont élevées (par exemple,  $S(1,2), S(3,4) > 0.7$ ), tandis que les synergies croisées telles que  $S(1,3)$ ,  $S(1,4)$ ,  $S(2,3)$  et  $S(2,4)$  restent faibles (typiquement  $< 0.2$ ). Dans ce cas, la mise à jour itérative renforce fortement les pondérations internes, de sorte que les valeurs  $\omega_{1,2}$  et  $\omega_{3,4}$  atteignent des niveaux élevés (par exemple, entre 0.6 et 0.8), tandis que les liaisons entre les groupes restent faibles (inférieures à 0.1). Si l'on ajoute un bruit additif modeste, par exemple  $\xi_{i,j}(t) \in [-0.02, +0.02]$ , l'effet sur la dynamique demeure marginal. En d'autres termes, la structure de clusters constituée par les paires  $\{\mathcal{E}_1, \mathcal{E}_2\}$  et  $\{\mathcal{E}_3, \mathcal{E}_4\}$  reste stable, démontrant une **résilience** élevée du SCN face aux perturbations faibles.

#### Exemple de Sensibilité :

En revanche, considérons un scénario dans lequel le SCN comprend cinq entités  $\{\mathcal{E}_1, \dots, \mathcal{E}_5\}$  et que la synergie interne d'un potentiel regroupement, par exemple  $S(1,2) \approx 0.3$ , est modérée et ne diffère pas significativement des synergies externes, qui se situeraient dans un intervalle approximatif de 0.1 à 0.2. Dans ce contexte, l'injection d'un bruit d'amplitude supérieure, disons  $\xi_{i,j}(t) \in [-0.05, +0.1]$ , peut provoquer des fluctuations plus importantes. La valeur de  $\omega_{1,2}(t)$  peut alors chuter de valeurs stables initialement situées autour de 0.25–0.3 à des niveaux inférieurs à 0.15. Une telle réduction critique de la pondération interne conduit à la désagrégation du cluster potentiel, illustrant une **sensibilité** marquée aux perturbations. Ce cas démontre que lorsque la marge de différenciation entre les synergies internes et externes est faible, le SCN devient fragile et vulnérable à des fluctuations aléatoires, même si celles-ci ne sont pas très élevées.

#### Cas de Multi-Attracteurs :

Dans certains systèmes comportant plusieurs attracteurs, comme discuté en section 2.3.2.1, l'injection de bruit peut jouer un rôle double. D'un côté, un bruit de faible amplitude peut contribuer à maintenir la flexibilité du système en permettant des fluctuations autour d'un attracteur stable.

D'un autre côté, un bruit persistant, même modéré, peut suffire à faire sortir le système de son bassin d'attraction initial, entraînant ainsi une transition vers une autre configuration attractrice. Cette capacité à basculer entre différents états stables souligne la dualité du rôle du bruit : il favorise à la fois l'exploration de nouvelles configurations et, en excès, il peut perturber la convergence vers un état stable.

### Synthèse et Conclusion :

La réaction d'un SCN à l'injection de bruit s'exprime sur un continuum allant de la **résilience** à la **sensibilité**. Lorsque l'amplitude du bruit  $\xi_{i,j}(t)$  est faible par rapport à la force corrective  $\eta [S(i,j) - \tau \omega_{i,j}(t)]$ , le système maintient ses structures auto-organisées malgré des perturbations mineures. En revanche, si le bruit atteint ou dépasse l'ordre de grandeur de ce terme de correction, les fluctuations peuvent altérer significativement les pondérations, conduisant à une désorganisation des clusters. Ainsi, la stabilité globale du SCN dépend de la comparaison précise entre ces deux contributions. Ce phénomène souligne l'importance de paramétrer soigneusement les valeurs de  $\eta$  et  $\tau$  pour assurer une dynamique robuste, comme évoqué dans les sections 2.3.1.3 et 2.3.2.1. En conclusion, le SCN peut démontrer une forte **résilience** en conservant ses regroupements lorsque le bruit est suffisamment faible, tandis qu'une **sensibilité** accrue, due à un bruit trop important ou à une faible différenciation entre synergies internes et externes, peut empêcher la stabilisation de structures cohérentes.

#### 2.3.4.3. Introduction à l'idée qu'un certain niveau de bruit peut parfois *aider* l'exploration (analogie avec le recuit simulé)

Dans le cadre du **Synergistic Connection Network (SCN)**, l'injection d'un élément de **bruit** peut, de manière paradoxale, améliorer la capacité du système à explorer l'espace des configurations, en l'aidant à échapper à des minima locaux qui ne sont pas nécessairement optimaux. Cette idée s'inspire du **recuit simulé** (simulated annealing) en optimisation, où l'on autorise initialement des fluctuations aléatoires significatives afin d'éviter un piégeage prématuré dans des configurations sous-optimales, avant de réduire progressivement l'intensité de ces perturbations pour permettre la stabilisation du système dans un état attracteur de haute qualité.

Considérons, à titre d'exemple, la règle de mise à jour additive fondamentale du SCN qui s'exprime sous la forme

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i,j) - \tau \omega_{i,j}(t)],$$

où  $\eta > 0$  est le **taux d'apprentissage** et  $\tau > 0$  représente le **coefficient de décroissance**. Pour intégrer l'effet de la **perturbation stochastique**, on modifie cette équation en ajoutant un terme aléatoire  $\xi_{i,j}(t)$  qui est souvent modélisé par une distribution gaussienne de moyenne nulle et de variance dépendante d'une **température**  $T(t)$  :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i,j) - \tau \omega_{i,j}(t)] + \xi_{i,j}(t), \quad \xi_{i,j}(t) \sim \mathcal{N}(0, \sigma^2(T(t))).$$

Au début de l'apprentissage, lorsque la **température**  $T(0)$  est élevée, l'amplitude de  $\xi_{i,j}(t)$  est grande, ce qui permet au réseau d'explorer un vaste espace de configurations. Cette phase d'exploration est essentielle pour éviter que le SCN ne se retrouve bloqué dans un minimum local qui pourrait être sous-optimal du point de vue de la **cohésion interne**. Au fil des itérations, un

**refroidissement progressif** de  $T(t)$  est appliqué, ce qui réduit graduellement la variance  $\sigma^2(T(t))$  du bruit. Ainsi, les fluctuations deviennent moins importantes et le système est amené à converger vers un arrangement plus stable, de sorte que, finalement, lorsque  $T(t)$  tend vers zéro, l'évolution de  $\omega_{i,j}(t)$  est dominée par le terme déterministe  $\eta [S(i,j) - \tau \omega_{i,j}(t)]$ .

L'analogie avec le **recuit simulé** se trouve dans la stratégie adoptée : initialement, le système autorise des sauts aléatoires – pouvant temporairement augmenter la valeur de la fonction « énergie » ou, dans notre cas, modifier les pondérations de manière non optimale – pour ensuite réduire progressivement ces fluctuations afin de "figer" la configuration dans un minimum qui est supposé être de qualité supérieure. On peut résumer ce mécanisme par l'idée qu'un certain niveau de **bruit** agit comme un moyen d'évasion des minima locaux et, par conséquent, d'**exploration** accrue du paysage de la **synergie**.

Les avantages d'une telle approche résident dans la capacité du système à éviter un piégeage précoce, favorisant ainsi l'émergence de regroupements (clusters) qui présentent une meilleure **cohésion interne**. En revanche, si le bruit persiste à un niveau trop élevé, il risque de perturber la convergence, empêchant le SCN de stabiliser ses liens et, par conséquent, de former des structures cohérentes. Ce compromis met en évidence la nécessité d'un **décroissement progressif** de la température, analogue à la planification de recuit dans le recuit simulé, afin de passer d'une phase d'exploration à une phase de **stabilisation**.

En conclusion, l'injection d'un bruit contrôlé dans un SCN peut être bénéfique en permettant au réseau d'explorer un plus grand nombre de configurations, d'éviter de se figer prématurément dans un minimum local et, finalement, d'aboutir à une structure globale plus optimale. Cette stratégie, inspirée du **recuit simulé**, souligne l'importance de paramétrer adéquatement les niveaux de bruit et leur décroissance (via la température  $T(t)$ ) pour concilier **exploration** et **stabilisation** dans le processus d'auto-organisation du **Deep Synergy Learning**.

### 2.3.5. Limites Théoriques et Questions Non Résolues

Les sections précédentes (2.3.1 à 2.3.4) ont présenté divers aspects de la **stabilité** dans un SCN, de la formation de clusters et des phénomènes de multi-attracteurs ou d'oscillations. Toutefois, la **théorie** du Deep Synergy Learning (DSL), envisagée comme un *système dynamique*, demeure largement **incomplète**. Il existe un ensemble d'**hypothèses** simplificatrices sous-jacentes et de **questions non résolues** quant à l'extension des modèles (apparition/disparition d'entités, synergie n-aire généralisée, etc.). La section 2.3.5 synthétise ces points, ouvrant la voie aux développements ultérieurs (notamment chapitres 9 et 12).

#### 2.3.5.1. Quelles hypothèses fortes sont faites (ex. synergie binaire, stationnarité, etc.)

Dans le cadre du Deep Synergy Learning, l'analyse théorique repose sur un ensemble d'hypothèses fortes qui permettent de simplifier l'étude de la dynamique du réseau, en particulier pour établir des résultats concernant la convergence, la stabilité et l'émergence de clusters. Ces hypothèses concernent principalement la **stationnarité** de la fonction de synergie, les **bornes** sur les pondérations, la forme de la **règle de mise à jour** ainsi que la nature même de la synergie.

La première hypothèse concerne la **stationnarité** (ou quasi-stationnarité) de la fonction de synergie. En effet, on postule souvent que la fonction  $S(i, j)$  ne varie pas de manière brusque ou aléatoire au cours du temps. Autrement dit, il est supposé que  $S(i, j)$  peut être considérée comme constante ou, à tout le moins, comme évoluant de manière suffisamment régulière. Cette hypothèse se formalise soit par le fait que  $S(i, j)$  dépend uniquement de paramètres invariants, tels que les représentations fixes  $\mathbf{x}_i$  et  $\mathbf{x}_j$  des entités, soit par la supposition que, lorsqu'une dépendance aux états internes  $\mathbf{s}_i$  est prise en compte, la fonction  $S(i, j)$  évolue à un rythme plus lent que les mises à jour des pondérations. Formellement, on peut exprimer cette hypothèse en admettant que, pour tout  $t$  et pour toute paire  $(i, j)$ , on a

$$S(i, j, t) \approx S(i, j),$$

ou bien que la variation temporelle, notée  $\Delta S(i, j, t)$ , est négligeable par rapport aux variations des poids.

Une autre hypothèse forte est celle concernant la **bornitude** des pondérations. On suppose qu'il existe un intervalle borné, souvent noté  $[0, \omega_{\max}]$ , dans lequel chaque  $\omega_{i,j}(t)$  évolue. Cette contrainte empêche théoriquement la divergence des poids et facilite l'application de théorèmes classiques tels que le théorème de point fixe de Brouwer. On exprime cette hypothèse par l'inégalité

$$0 \leq \omega_{i,j}(t) \leq \omega_{\max}, \quad \forall i, j, \forall t,$$

ce qui permet également de considérer l'espace des configurations comme compact et convexe, une condition essentielle pour la convergence.

Une troisième hypothèse réside dans la **forme de mise à jour** des pondérations. Dans de nombreux travaux sur le DSL, la règle de mise à jour est adoptée sous une forme additive, généralement donnée par

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i, j) - \tau \omega_{i,j}(t)],$$

où  $\eta > 0$  est le taux d'apprentissage et  $\tau > 0$  est le coefficient de décroissance. Cette forme linéaire simplifie l'analyse, notamment la linéarisation locale autour d'un point fixe, permettant de montrer que, dans le cas stationnaire, le point fixe pour une liaison donnée est

$$\omega_{i,j}^* = \frac{S(i, j)}{\tau}.$$

L'hypothèse d'une mise à jour additive permet ainsi d'établir, par linéarisation, des conditions de stabilité via l'étude de la matrice Jacobienne. Cependant, il convient de noter que dans des versions plus avancées, des mises à jour de nature multiplicative ou intégrant des couplages n-aires apparaissent, ce qui rend la dynamique plus complexe et nécessite de relâcher cette hypothèse de linéarité.

Enfin, certaines simplifications interviennent dans la **nature** même de la synergie. Pour rendre l'analyse plus tractable, il est parfois supposé que  $S(i, j)$  est de nature **binaire**, c'est-à-dire que  $S(i, j) \in \{0, 1\}$ . Une telle hypothèse permet de transformer la mesure continue de synergie en un indicateur discret qui détermine simplement si une connexion entre deux entités est présente ou non. Bien que cette approche puisse être très utile pour certaines applications (notamment lorsque

l'on souhaite obtenir des structures denses et clairement définies), elle perd en finesse et en nuance, puisque la gradation de la synergie est alors remplacée par un seuil abrupt.

En synthèse, l'ensemble de ces hypothèses – la stationnarité (ou quasi-stationnarité) de  $S(i, j)$ , la contrainte de bornitude des poids dans  $[0, \omega_{\max}]$ , l'adoption d'une règle de mise à jour additive, ainsi que la simplification de  $S(i, j)$  en une variable binaire ou en une mesure continue régularisée – constitue un cadre restreint mais opératoire pour l'analyse théorique du DSL. Ces hypothèses facilitent l'étude de la convergence, la détection de points fixes, ainsi que l'identification des clusters, en rendant l'opérateur dynamique  $F$  suffisamment « maniable » pour l'application de méthodes analytiques classiques. Toutefois, il est important de noter que dans des applications réelles, où les entités peuvent apparaître et disparaître, ou lorsque des interactions plus complexes (comme la synergie n-aire) sont à l'œuvre, il sera nécessaire de relâcher certaines de ces hypothèses pour capter toute la richesse du comportement du système. Les sections suivantes, notamment 2.3.5.2 et 2.3.5.3, aborderont ces ouvertures et exploreront des généralisations qui permettront de modéliser des phénomènes plus complexes tout en gardant une base théorique solide.

### 2.3.5.2. Ouvertures vers une gestion plus dynamique (entités qui apparaissent ou disparaissent)

Dans le cadre du **Deep Synergy Learning**, il est fondamental de considérer des scénarios où le nombre d'**entités** n'est pas fixe, mais évolue au fil du temps. Cette ouverture vers une gestion dynamique des entités reflète la réalité de nombreux systèmes complexes, tels que les réseaux de capteurs, les plateformes sociales ou les systèmes de robotique coopérative, dans lesquels de nouveaux éléments peuvent apparaître tandis que d'autres disparaissent ou deviennent inactifs. Cette dimension temporelle supplémentaire exige une extension du modèle de mise à jour des **pondérations**  $\{\omega_{i,j}\}$  de sorte à permettre l'adaptation du **\*\*Synergistic Connection Network (SCN)** face à l'évolution du nombre d'unités.

#### A. Motivations et enjeux

Dans de nombreux domaines, la dynamique du système ne se limite pas à l'évolution des **liaisons** entre entités statiques, mais doit également intégrer la **naissance** et la **disparition** d'unités. Par exemple, dans un réseau de capteurs IoT, l'ajout de nouveaux dispositifs ou la défaillance de certains capteurs modifie la structure globale du réseau, de même qu'un réseau social, soumis à des flux continus d'inscriptions et de désinscriptions, présente une topologie en constante évolution. En robotique coopérative, la flotte de robots peut s'agrandir ou se réduire selon les besoins opérationnels ou les incidents techniques. Pour que le SCN puisse modéliser de telles situations, il faut autoriser la structure de **pondérations** à évoluer non seulement en réponse aux fluctuations de la **synergie**  $S(i, j)$ , mais également en fonction du nombre d'**entités** présentes.

#### B. Approche mathématique pour la gestion dynamique

Si l'on note  $\Omega(t)$  l'ensemble des pondérations pour un réseau comportant initialement  $n$  entités, alors  $\Omega(t)$  est un vecteur dont les composantes sont données par

$$\Omega(t) = (\omega_{i,j}(t) \mid i, j \in \{1, \dots, n\}).$$



Lorsqu'une nouvelle entité, que l'on notera  $\mathcal{E}_{n+1}$ , rejoint le réseau, l'espace d'état des pondérations s'étend naturellement. En effet, il devient nécessaire d'introduire de nouvelles variables, par exemple  $\omega_{n+1,j}(t)$  et  $\omega_{j,n+1}(t)$  pour  $j = 1, \dots, n$ , afin de décrire les interactions entre la nouvelle entité et celles déjà existantes. L'opérateur de mise à jour, qui était auparavant défini sur l'espace  $\mathcal{X}_n$ , doit être adapté pour opérer sur  $\mathcal{X}_{n+1}$ . Une telle extension se traduit par l'initialisation de ces nouvelles pondérations, par exemple par

$$\omega_{n+1,j}(0) \approx 0 \quad \text{et} \quad \omega_{j,n+1}(0) \approx 0,$$

afin de laisser à la dynamique du SCN le temps de déterminer, via la règle

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i,j) - \tau \omega_{i,j}(t)],$$

la pertinence des liaisons entre la nouvelle entité et le reste du réseau. Inversement, lorsque qu'une entité  $\mathcal{E}_k$  disparaît ou devient inactif, on suppose que l'on met à zéro les pondérations correspondantes, c'est-à-dire que pour tout  $j$ , on fixe

$$\omega_{k,j}(t) = \omega_{j,k}(t) = 0,$$

et l'espace des pondérations se contracte de manière appropriée de  $\mathcal{X}_n$  vers  $\mathcal{X}_{n-1}$ . Cette procédure de retrait impose un rééquilibrage du réseau, car les liens qui dépendaient de l'entité disparue devront être redistribués parmi les entités restantes, pouvant entraîner une réorganisation de la structure des **clusters**.

### *C. Conséquences sur la stabilité et la formation de clusters*

L'arrivée ou le départ d'**entités** constitue une perturbation dans la dynamique globale du SCN. Si, par exemple, une nouvelle entité  $\mathcal{E}_{n+1}$  manifeste une forte synergie avec certains sous-groupes existants, cela peut conduire à une reconfiguration des clusters. Une partie du cluster initial peut se scinder pour intégrer la nouvelle entité, ou bien les pondérations existantes peuvent être ajustées pour optimiser la cohésion globale. Dans le cas d'une entité qui se retire, l'équilibre du réseau est modifié : le cluster auquel elle appartenait peut se morceler ou fusionner avec un autre, selon l'importance de sa contribution initiale au niveau des **synergies** internes.

D'un point de vue algorithmique, deux stratégies principales peuvent être envisagées pour gérer ces fluctuations de dimension. La première consiste à adopter une approche de **continuité stricte**, dans laquelle l'insertion ou le retrait d'une entité s'effectue de manière incrémentale, permettant aux pondérations existantes de se réajuster progressivement sans nécessiter une réinitialisation complète du système. La seconde stratégie, appelée **réinitialisation partielle**, consiste à réinitialiser partiellement certaines pondérations lorsque le changement de dimension est jugé majeur, afin de faciliter la recomposition de la dynamique du réseau.

### *D. Exemples pratiques et implications*

Considérons un scénario typique dans un réseau de capteurs où de nouveaux dispositifs sont ajoutés régulièrement. Dès l'insertion d'une nouvelle entité  $\mathcal{E}_{n+1}$ , les nouvelles pondérations  $\omega_{n+1,j}$  sont initialisées à une valeur proche de zéro. La règle de mise à jour,

$$\omega_{n+1,j}(t+1) = \omega_{n+1,j}(t) + \eta [S(n+1,j) - \tau \omega_{n+1,j}(t)],$$

détermine ensuite la manière dont  $\mathcal{E}_{n+1}$  se connecte aux entités existantes, ce qui peut entraîner la formation de nouveaux clusters ou la fusion de clusters existants pour mieux intégrer la nouvelle information. Inversement, dans un réseau social, la disparition d'un utilisateur  $\mathcal{E}_k$  implique la mise à zéro de toutes les pondérations associées, ce qui peut provoquer une reconfiguration du cluster auquel il appartenait, modifiant ainsi la partition finale du réseau.

### *E. Limites théoriques et perspectives*

L'ouverture vers une gestion dynamique des entités pose cependant plusieurs défis théoriques. En effet, lorsque la dimension de l'espace des pondérations n'est plus fixe, les outils classiques de la théorie des points fixes, qui reposent sur la compacité et la convexité de cet espace, deviennent partiellement inopérants. Le passage de  $\mathcal{X}_n$  à  $\mathcal{X}_{n+1}$  ou  $\mathcal{X}_{n-1}$  entraîne une modification de la structure même de l'opérateur  $F$ , ce qui nécessite le développement de méthodes incrémentales pour analyser la stabilité du système en évolution. De plus, l'inflation du nombre de liaisons, qui croît typiquement en  $O(n)$  pour chaque nouvelle entité, peut entraîner une augmentation significative des coûts computationnels, ce qui impose souvent l'emploi de techniques de **sparsification** pour garantir la praticabilité du modèle.

Ces défis ouvrent des perspectives intéressantes pour le développement du **Deep Synergy Learning**. Des recherches futures pourraient explorer des stratégies de réinitialisation partielle ou des méthodes adaptatives de recalibrage des pondérations afin de conserver un équilibre entre la stabilité du réseau et sa capacité à s'adapter aux évolutions du nombre d'entités. Des travaux ultérieurs, notamment ceux qui seront abordés dans le Chapitre 9, se pencheront sur ces questions en profondeur, intégrant des approches de **mise à jour incrémentale** et des algorithmes d'**apprentissage continu** qui tiennent compte de l'apparition et de la disparition des entités.

### *Conclusion (2.3.5.2)*

L'ouverture vers une gestion dynamique des entités, où le SCN peut croître avec l'arrivée de nouvelles unités ou se réduire en cas de disparition, constitue une extension essentielle du modèle de **Deep Synergy Learning**. Cette gestion implique une adaptation de la structure des **pondérations**  $\{\omega_{i,j}\}$  par l'élargissement ou la contraction de l'espace d'état, ce qui modifie l'opérateur de mise à jour  $F$ . Bien que cette approche rende l'analyse théorique plus complexe – notamment en remettant en question les hypothèses classiques de compacité et de stabilité dans un espace de dimension fixe – elle reflète de manière réaliste les dynamiques de systèmes évolutifs. Les stratégies envisagées, telles que l'insertion incrémentale ou la réinitialisation partielle, offrent des pistes prometteuses pour conserver la **stabilité** et la **cohérence** des clusters malgré la fluctuation du nombre d'entités. Ces perspectives seront approfondies dans les chapitres suivants, notamment dans le contexte de l'apprentissage continu et de la synergie multimodale, afin d'étendre le cadre théorique du DSL à des systèmes véritablement dynamiques.

### **2.3.5.3. Préfigure les Chapitres 9 et 12 sur l'apprentissage continu et la synergie n-aire**

Dans les sections précédentes, notamment en 2.3.5.1 et 2.3.5.2, nous avons établi un cadre théorique fondé sur des hypothèses fortes concernant la stationnarité de la **synergie**  $S(i, j)$  ainsi

que sur la gestion dynamique des entités dans un **Synergistic Connection Network (SCN)**. Il apparaît désormais indispensable d'envisager deux axes majeurs d'extension du modèle, axes qui seront développés de manière approfondie dans les **Chapitre 9** et **Chapitre 12** de cet ouvrage. La première orientation consiste en l'**apprentissage continu** (ou *lifelong learning*), tandis que la seconde porte sur l'extension de la synergie aux interactions **n-aires**. Ces deux axes permettent d'élargir le domaine d'application du DSL et de prendre en compte des scénarios où la dynamique des entités et la complexité des interactions dépassent le cadre strictement binaire et stationnaire.

Dans le contexte de l'**apprentissage continu**, le SCN se trouve confronté à des environnements évolutifs, où de nouvelles données ou de nouvelles entités apparaissent, tandis que d'autres disparaissent ou deviennent inactives. Au lieu de supposer un nombre fixe d'entités – hypothèse qui permettait d'appliquer des théorèmes de convergence dans un espace compact et convexe – il faut maintenant envisager une extension de la dynamique du réseau. Plus précisément, si l'on note initialement par

$$\Omega(t) = (\omega_{1,2}(t), \omega_{1,3}(t), \dots, \omega_{n-1,n}(t))$$

la configuration des pondérations pour un ensemble de  $n$  entités, l'arrivée d'une nouvelle entité  $\mathcal{E}_{n+1}$  entraîne l'extension de l'espace d'état à

$$\Omega'(t) = (\omega_{1,2}(t), \dots, \omega_{n-1,n}(t), \omega_{n+1,1}(t), \omega_{n+1,2}(t), \dots, \omega_{n+1,n}(t)).$$

La règle de mise à jour demeure alors de la forme

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i,j) - \tau \omega_{i,j}(t)],$$

mais l'opérateur  $F$  doit désormais être défini sur un espace de dimension variable. Ce processus d'intégration progressive de nouvelles entités nécessite des mécanismes spécifiques pour éviter le phénomène de « retrait catastrophique » (catastrophic forgetting) et pour permettre une adaptation continue sans réinitialiser l'ensemble du réseau. C'est précisément l'objet du **Chapitre 9**, qui développera des stratégies d'apprentissage incrémental permettant de concilier la préservation des connaissances acquises avec l'adaptation à de nouveaux contextes, ainsi que des ajustements paramétriques dynamiques de  $\eta$  et  $\tau$  pour optimiser la stabilité dans un environnement évolutif.

En parallèle, la généralisation de la synergie aux interactions **n-aires** représente une deuxième extension fondamentale. Jusqu'ici, le DSL s'est principalement appuyé sur une fonction de synergie définie pour des paires d'entités, c'est-à-dire  $S(i,j)$ . Or, de nombreuses applications requièrent de modéliser des interactions collectives impliquant trois, quatre ou un nombre plus grand d'entités simultanément. Dans ce cadre, il convient d'introduire une fonction générale

$$S(i_1, i_2, \dots, i_k),$$

qui mesure la **co-information** ou la **synergie collective** d'un groupe d'entités  $\{\mathcal{E}_{i_1}, \mathcal{E}_{i_2}, \dots, \mathcal{E}_{i_k}\}$ . Cette généralisation conduit naturellement au passage d'un graphe pondéré à un **hypergraphe**, dans lequel les hyper-liens représentent des interactions entre plusieurs nœuds de manière intrinsèquement non binaire. Formellement, il devient alors pertinent de définir des pondérations telles que

$$\omega_{(i_1, i_2, \dots, i_k)}(t+1) = \omega_{(i_1, i_2, \dots, i_k)}(t) + \eta [S(i_1, i_2, \dots, i_k) - \tau \omega_{(i_1, i_2, \dots, i_k)}(t)],$$

ce qui élargit considérablement le champ d'application du DSL. Le **Chapitre 12** se consacrera à formaliser ce concept de synergie **n-aire** et à étudier l'impact de ces interactions multiples sur l'auto-organisation et la stabilité globale du réseau. L'analyse portera notamment sur la manière dont ces hyper-liens influencent la formation de clusters et sur les nouveaux phénomènes dynamiques qui émergent dans un hypergraphe, en intégrant des aspects de co-information collective et d'interactions non linéaires plus sophistiquées.

En résumé, l'intégration de l'**apprentissage continu** et de la **synergie n-aire** constitue une avancée majeure qui préfigure un DSL de nouvelle génération, capable de s'adapter de manière incrémentale aux évolutions de son environnement tout en prenant en compte des interactions multi-entités. Ces deux axes d'extension ouvrent des perspectives particulièrement intéressantes pour des applications dans des domaines tels que les réseaux de capteurs, la robotique multi-agents ou encore les systèmes neuronaux inspirés de la biologie, où la flexibilité et la complexité des interactions sont essentielles. Les **Chapitre 9** et **Chapitre 12** développeront ces idées en profondeur, en proposant des modèles mathématiques rigoureux ainsi que des approches algorithmiques adaptées pour garantir la **stabilité** et la **cohérence** dans des environnements dynamiques et multidimensionnels.

Ainsi, cette section sert de pont vers une compréhension plus large du DSL, en préparant le terrain pour une gestion avancée de l'**apprentissage continu** et l'extension de la **synergie** au-delà du cadre binaire traditionnel, contribuant à l'émergence de réseaux véritablement **adaptatifs** et **complexes**.

## 2.4. Raccords avec la Physique Statistique et la Théorie des Systèmes Dynamiques

Le **Deep Synergy Learning** (DSL), vu sous l'angle d'un **réseau**  $\{\omega_{i,j}\}$  évoluant sous des règles non linéaires, peut être relié à des approches classiques de la **physique statistique** (modèles de spins, énergies) et de la **théorie des systèmes dynamiques** (discrets ou continus). La section 2.4 initie ce rapprochement, en commençant (2.4.1) par l'interprétation de l'équation de mise à jour  $\omega_{i,j}(t+1)$  comme un **système dynamique discret** et ses variantes continues, puis (2.4.2) et (2.4.3) feront des parallèles explicites avec les modèles de spin et la notion d'énergie ou de fonction potentielle.

### 2.4.1. Systèmes Dynamiques Discrets ou Continus

On a souvent présenté la mise à jour du SCN (voir 2.2.2) sous forme discrète :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta[S(i,j)\tau \omega_{i,j}(t)],$$

mais il est tout à fait possible de concevoir une version **continue** (2.4.1.2) ou même de tenter des analogies directes avec les **systèmes dynamiques** utilisés en physique statistique (2.4.2). Commençons (2.4.1.1) par voir en quoi  $\omega_{i,j}(t+1) = F(\omega_{i,j}(t), \dots)$  s'interprète déjà comme un **système dynamique discret** et comment les outils de cette théorie peuvent s'appliquer.

#### 2.4.1.1. Équation de mise à jour $\omega_{i,j}(t+1)$ vue comme un système dynamique discrétisé

Dans la théorie des **systèmes dynamiques**, un modèle discret se représente typiquement par l'équation

$$\mathbf{x}(t+1) = F(\mathbf{x}(t)), \quad \mathbf{x} \in \mathbb{R}^D,$$

où chaque itération convertit l'état  $\mathbf{x}(t)$  en  $\mathbf{x}(t+1)$  au moyen d'une **application** (ou **opérateur**)  $F$ . Dans le cadre du **Deep Synergy Learning** (DSL), l'état du réseau est encapsulé dans le vecteur  $\mathbf{\Omega}(t)$  qui regroupe l'ensemble des **pondérations**  $\{\omega_{i,j}(t)\}$  et, éventuellement, d'autres variables d'état telles que les **états internes** des entités. On peut donc écrire

$$\mathbf{\Omega}(t+1) = F(\mathbf{\Omega}(t)),$$

où l'opérateur  $F$  intègre la **règle de mise à jour** du DSL, que l'on exprime généralement sous la forme

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta[S(i,j;\mathbf{\Omega}(t)) - \tau \omega_{i,j}(t)].$$

Dans cette équation,  $S(i,j;\mathbf{\Omega}(t))$  représente la **synergie effective** évaluée à l'itération  $t$  et peut dépendre non seulement des représentations intrinsèques des entités, telles que leurs **vecteurs caractéristiques**  $\mathbf{x}_i$  ou leurs **états internes**  $\mathbf{s}_i$ , mais aussi de l'ensemble des pondérations  $\omega_{p,q}(t)$  existant dans le réseau. Le paramètre  $\eta > 0$  est le **taux d'apprentissage** qui détermine la vitesse d'évolution du système, tandis que  $\tau > 0$  représente le **coefficient de décroissance**, essentiel pour éviter une croissance non bornée des pondérations et pour assurer la régulation du processus.

Si l'on considère un réseau composé de  $n$  entités et que les liens sont orientés, le nombre total de pondérations possibles est de l'ordre de  $n(n - 1)$ ; par conséquent, le vecteur d'état  $\mathbf{\Omega}(t)$  appartient à un espace de dimension  $D \approx n(n - 1)$ . L'opérateur  $F$  agit donc sur un espace de grande dimension, et sa définition se fait de manière à ce que, pour toute condition initiale  $\mathbf{\Omega}(0)$  appartenant à un ensemble borné  $\mathcal{X} \subset \mathbb{R}^D$  (par exemple, chaque  $\omega_{i,j}$  étant contraint à l'intervalle  $[0, \omega_{\max}]$ ), la suite  $\{\mathbf{\Omega}(t)\}$  demeure dans  $\mathcal{X}$ . Autrement dit, la dynamique se décrit par

$$\mathbf{\Omega}(t + 1) = F(\mathbf{\Omega}(t)), \quad \text{avec} \quad \mathbf{\Omega}(t) \in \mathcal{X},$$

et l'**orbite** ou la **trajectoire** du système est définie par l'ensemble  $\{\mathbf{\Omega}(0), \mathbf{\Omega}(1), \mathbf{\Omega}(2), \dots\}$ .

L'intérêt de cette formalisation réside dans la capacité à analyser, du point de vue de la **stabilité** et de la **convergence**, la dynamique des poids. En effet, comme indiqué dans la section 2.3.1.3, la stabilité locale d'un point fixe  $\mathbf{\Omega}^*$  repose sur l'étude de la linéarisation de  $F$  autour de ce point. Lorsque la norme de la **matrice Jacobienne**  $DF(\mathbf{\Omega}^*)$  est strictement inférieure à 1, c'est-à-dire

$$\|DF(\mathbf{\Omega}^*)\| < 1,$$

le point fixe  $\mathbf{\Omega}^*$  se révèle stable ; sinon, des phénomènes tels que des cycles, des multi-attracteurs ou même des comportements pseudo-chaotiques peuvent apparaître (voir également la discussion des sections 2.3.2 et 2.3.3).

Une illustration théorique simple de la dynamique consiste à considérer le cas stationnaire où  $S(i, j; \mathbf{\Omega}(t))$  est approximativement constant. Dans ce cas, la règle de mise à jour se simplifie en

$$\omega_{i,j}(t + 1) = (1 - \eta \tau) \omega_{i,j}(t) + \eta S(i, j),$$

ce qui, sous l'hypothèse de stabilité, conduit à la convergence des poids vers le point fixe

$$\omega_{i,j}^* = \frac{S(i, j)}{\tau}.$$

Cette condition de stabilité, qui se traduit par l'inégalité

$$|1 - \eta \tau| < 1,$$

implique que, dans un cas simple, le produit  $\eta \tau$  doit satisfaire  $0 < \eta \tau < 2$ . Les **paramètres**  $\eta$  et  $\tau$  sont donc cruciaux pour la bonne marche du système : un  $\eta$  trop élevé peut induire des oscillations, tandis qu'un  $\tau$  trop grand risque d'amener les poids à s'écraser, limitant ainsi l'amplitude des interactions.

Par analogie avec des modèles de **spins** tels que le modèle d'**Ising** ou le réseau de **Hopfield**, qui sont souvent mis à jour de manière itérative (par exemple, via l'algorithme Metropolis-Hastings), le DSL peut être vu comme un système dynamique discret opérant sur le vecteur  $(\omega_{1,2}, \omega_{1,3}, \dots, \omega_{n-1,n})$ . Cette analogie s'étendra dans les sections 2.4.2 et 2.4.3, où seront établis des liens supplémentaires avec la **fonction d'énergie** et les transitions de phase, offrant ainsi une vision énergétique de la dynamique du DSL.

Pour formaliser cette approche, nous pouvons énoncer le théorème suivant :

### Théorème (Existence et Itération de la Dynamique Discrète)

Soit  $\mathcal{X} \subset \mathbb{R}^D$  un ensemble **compact** (par exemple, chaque  $\omega_{i,j}$  étant borné dans  $[0, \omega_{\max}]$ ). Soit  $F: \mathcal{X} \rightarrow \mathcal{X}$  une application **continue** qui définit la mise à jour du vecteur d'état selon

$$\mathbf{\Omega}(t+1) = F(\mathbf{\Omega}(t)).$$

Alors, pour toute condition initiale  $\mathbf{\Omega}(0) \in \mathcal{X}$ , la suite  $\{\mathbf{\Omega}(t)\}_{t \geq 0}$  reste dans  $\mathcal{X}$  et, par le **théorème de Brouwer** (ou Schauder), il existe au moins un **point fixe**  $\mathbf{\Omega}^* \in \mathcal{X}$  tel que

$$\mathbf{\Omega}^* = F(\mathbf{\Omega}^*).$$

La stabilité ou la multiplicité de ce point fixe dépendra des propriétés locales de l'opérateur  $F$ , notamment de la norme  $\|DF(\mathbf{\Omega}^*)\|$ .

En pratique, le fait de **borner** les valeurs de  $\omega_{i,j}$  (par clipping ou par l'application de règles de **parsimonie**) et d'adopter une règle de mise à jour continue assure que la trajectoire  $\mathbf{\Omega}(0) \rightarrow \mathbf{\Omega}(1) \rightarrow \mathbf{\Omega}(2) \rightarrow \dots$  reste bien définie dans l'espace  $\mathcal{X}$ . Ainsi, la dynamique du DSL s'inscrit pleinement dans le cadre standard des systèmes dynamiques discrets, et la stabilité des clusters ainsi formés dépend essentiellement des propriétés de  $F$ .

### Conclusion

L'équation de mise à jour

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i,j; \mathbf{\Omega}(t)) - \tau \omega_{i,j}(t)],$$

se lit comme une représentation typique d'un **système dynamique discret** opérant sur le vecteur  $\mathbf{\Omega}(t)$  dans un espace de dimension  $D \approx n(n-1)$ . L'opérateur  $F: \mathcal{X} \rightarrow \mathcal{X}$  qui en résulte, supposé continu et défini sur un ensemble compact (par exemple, par des bornes imposées aux  $\omega_{i,j}$ ), garantit par le théorème de Brouwer l'existence d'au moins un point fixe. La stabilité locale de ce point fixe se mesure par l'analyse de la **matrice Jacobienne**  $DF(\mathbf{\Omega}^*)$  et l'exigence que  $\|DF(\mathbf{\Omega}^*)\| < 1$ . Par ailleurs, l'analogie avec les modèles de spins (Ising, Hopfield) et l'approche énergétique permet de mieux comprendre les transitions de phase et les dynamiques complexes observées dans le DSL. Dans la section suivante (2.4.1.2), nous aborderons la variante continue de ce système, en formulant une équation différentielle qui offre un nouveau pont entre le DSL et la modélisation des systèmes hors équilibre en physique et en biologie.

#### 2.4.1.2. Variante continue : $\frac{d\omega_{ij}}{dt} = \eta [S(i,j) - \tau \omega_{ij}]$

Dans la plupart des présentations du **Deep Synergy Learning (DSL)**, la mise à jour des pondérations se réalise en temps discret par des itérations successives, ce qui conduit à une équation de la forme

$$\omega_{ij}(t + 1) = \omega_{ij}(t) + \eta [S(i, j) - \tau \omega_{ij}(t)].$$

Cependant, il est tout à fait envisageable de modéliser l'évolution des pondérations par une **équation différentielle** en temps continu. Pour ce faire, on part de la différence discrète exprimée par

$$\frac{\omega_{ij}(t + 1) - \omega_{ij}(t)}{\Delta t} = \eta [S(i, j) - \tau \omega_{ij}(t)],$$

et, en posant  $\Delta t = 1$  puis en faisant tendre  $\Delta t$  vers zéro, la limite conduit à l'équation différentielle

$$\frac{d \omega_{ij}}{dt} = \eta [S(i, j) - \tau \omega_{ij}(t)].$$

Cette formulation permet d'envisager la dynamique des pondérations de manière « fluide » plutôt que par paliers discrets. Dans le cas où la **synergie**  $S(i, j)$  est considérée comme constante (ou stationnaire), l'équation différentielle devient linéaire du premier ordre :

$$\frac{d \omega_{ij}}{dt} = \eta [S(i, j) - \tau \omega_{ij}(t)].$$

Pour une condition initiale  $\omega_{ij}(0) = \omega_0$ , la solution générale s'obtient par intégration et s'écrit ainsi :

$$\omega_{ij}(t) = \frac{S(i, j)}{\tau} + \left( \omega_0 - \frac{S(i, j)}{\tau} \right) \exp(-\eta \tau t).$$

Cette solution démontre que, lorsque  $t \rightarrow +\infty$ , la pondération converge de manière exponentielle vers le point fixe

$$\omega_{ij}^* = \frac{S(i, j)}{\tau},$$

la vitesse de convergence étant régie par le produit  $\eta \tau$ . Ce résultat rejoint exactement la conclusion obtenue dans le modèle discret, tout en offrant une interprétation continue et en ouvrant la voie à l'utilisation d'outils analytiques propres aux **équations différentielles ordinaires (EDO)**.

Lorsque la **synergie**  $S(i, j)$  dépend non seulement de  $\mathbf{x}_i$  et  $\mathbf{x}_j$  mais également d'autres pondérations ou d'états internes (par exemple, lorsque  $S$  devient une fonction de l'ensemble  $\{\omega_{p,q}\}$  ou des états  $\mathbf{s}_k$ ), le système se généralise à un ensemble couplé d'équations différentielles :

$$\frac{d \omega_{ij}}{dt} = \eta [S(i, j; \{\omega_{p,q}\}) - \tau \omega_{ij}(t)], \quad \forall (i, j).$$

Le vecteur global  $\mathbf{\Omega}(t)$ , constitué de l'ensemble des pondérations, évolue alors dans un espace de dimension potentiellement élevée, souvent de l'ordre de  $n(n - 1)$  si les liens sont orientés. Les propriétés de **continuité** et de **Lipschitz** de la fonction  $S$  garantissent, par le théorème de **Cauchy-Lipschitz**, l'existence et l'unicité locale des solutions. La stabilité d'un point fixe  $\mathbf{\Omega}^*$  dans ce contexte se caractérise par l'étude de la linéarisation de l'opérateur  $F$  associé à la dynamique continue, qui se traduit par



$$\delta \mathbf{\Omega}'(t) = DF(\mathbf{\Omega}^*) \delta \mathbf{\Omega}(t),$$

où  $DF(\mathbf{\Omega}^*)$  désigne la **matrice Jacobienne** évaluée en l'équilibre. Si toutes les valeurs propres de cette Jacobienne ont des parties réelles strictement négatives, l'équilibre est localement attractif.

Il convient de noter que, dans la pratique, bien que le DSL soit souvent implémenté en temps discret avec un pas  $\Delta t$  très petit (par exemple, entre 0.01 et 0.1), l'analyse en temps continu offre un cadre conceptuel riche, permettant d'établir des analogies avec des modèles de **spins** et d'autres systèmes neuronaux (comme ceux de **Hopfield** ou de **Cohen-Grossberg**). Ce passage à une formulation continue rapproche ainsi le DSL des modèles de systèmes dynamiques continus rencontrés en physique et en biologie, enrichissant l'analyse théorique et facilitant l'étude des propriétés de convergence et de stabilité.

En somme, la variante continue du DSL s'exprime par

$$\frac{d \omega_{ij}}{dt} = \eta [S(i, j) - \tau \omega_{ij}],$$

ce qui, dans le cas stationnaire, implique que  $\omega_{ij}(t)$  converge exponentiellement vers le point fixe  $\omega_{ij}^* = \frac{S(i, j)}{\tau}$ . Dans le cas général, où  $S(i, j)$  dépend des pondérations ou des états internes, le système se complexifie en un ensemble couplé d'équations différentielles dont l'analyse repose sur des techniques classiques de linéarisation et d'étude de la stabilité par la Jacobienne. Cette formulation continue offre une perspective alternative et complémentaire à la dynamique discrète, et permet de mieux comprendre les analogies entre le DSL et les modèles neuronaux ou les systèmes de spins étudiés en physique mathématique.

#### 2.4.1.3. Conditions d'existence d'une solution fermée ou d'une approximation

Dans la version continue du **Deep Synergy Learning (DSL)**, la dynamique des pondérations  $\omega_{ij}(t)$  est régie par un système d'équations différentielles ordinaires (EDO) généralement écrit sous la forme

$$\frac{d \omega_{ij}}{dt} = \eta [S(i, j; \{\omega_{p,q}\}, \dots) - \tau \omega_{ij}(t)],$$

où  $\eta > 0$  représente le **taux d'apprentissage**,  $\tau > 0$  le **coefficient de décroissance** et  $S(i, j; \{\omega_{p,q}\}, \dots)$  décrit la **synergie effective** qui peut, selon les cas, dépendre de l'ensemble des pondérations  $\{\omega_{p,q}\}$  et d'autres variables (par exemple, les états internes des entités). L'objectif est de déterminer, d'une part, dans quels cas on peut obtenir une **solution fermée** (analytique) pour ces équations et, d'autre part, quelles méthodes théoriques et numériques permettent de prouver l'**existence**, l'**unicité** et la **qualité** des solutions lorsque la solution fermée n'est pas accessible.

##### A. Solutions fermées dans les cas simples

Lorsque la fonction de synergie  $S(i, j)$  est supposée **constante** ou **indépendante** de l'ensemble des pondérations  $\{\omega_{p,q}\}$ , l'équation se simplifie en une équation différentielle linéaire du premier ordre :

$$\frac{d \omega_{ij}}{dt} = \eta [S(i, j) - \tau \omega_{ij}(t)].$$

Cette équation est classique et se résout par séparation des variables ou par la méthode de variation des constantes. En imposant la condition initiale  $\omega_{ij}(0) = \omega_0$ , la solution analytique se trouve être

$$\omega_{ij}(t) = \frac{S(i, j)}{\tau} + \left( \omega_0 - \frac{S(i, j)}{\tau} \right) \exp(-\eta \tau t).$$

On constate ainsi que, pour  $t \rightarrow +\infty$ , la pondération converge exponentiellement vers le point fixe

$$\omega_{ij}^* = \frac{S(i, j)}{\tau},$$

ce qui confirme la validité de cette solution fermée dans le cas linéaire stationnaire. De même, si l'on suppose que  $S(i, j)$  dépend de manière linéaire des pondérations et reste stationnaire, le système se réduit à un système linéaire de la forme

$$\mathbf{x}'(t) = M \mathbf{x}(t) + \mathbf{b},$$

où l'on peut alors résoudre le système par diagonalisation de la matrice  $M$ . Ces cas simples illustrent que, sous certaines hypothèses restrictives, une **solution fermée** est effectivement accessible.

### *B. Cas non linéaire : $S$ dépendant de $\{\omega_{p,q}\}$*

Dans la plupart des scénarios pratiques du DSL, la synergie  $S(i, j; \{\omega_{p,q}(t)\}, \dots)$  intègre des **non-linéarités** dues à des fonctions de distance, des mesures de co-information ou à des mécanismes d'inhibition compétitive. Dans ces cas, l'équation devient un système couplé non linéaire :

$$\frac{d \omega_{ij}}{dt} = \eta [S(i, j; \{\omega_{p,q}(t)\}) - \tau \omega_{ij}(t)], \quad \forall (i, j).$$

Ce système, que l'on peut écrire sous la forme

$$\mathbf{\Omega}'(t) = \mathbf{F}(\mathbf{\Omega}(t)),$$

avec  $\mathbf{\Omega}(t) \in \mathbb{R}^D$  et  $D \approx n(n-1)$  dans le cas d'un réseau orienté, ne permet généralement pas d'obtenir une solution fermée. Cependant, si la fonction  $\mathbf{F}$  est **localement Lipschitz**, le théorème de **Cauchy–Lipschitz** (ou théorème d'existence et d'unicité) garantit qu'une solution unique existe pour chaque condition initiale  $\mathbf{\Omega}(0)$ . En pratique, l'intégration de telles équations non linéaires requiert le recours à des méthodes d'**approximation numérique** ou à des techniques d'analyse qualitative telles que la linéarisation autour d'un point fixe.

### *C. Méthodes numériques et linéarisation*

Pour étudier ou simuler la trajectoire de  $\omega_{ij}(t)$  dans le cas non linéaire, plusieurs schémas numériques peuvent être employés. Par exemple, la méthode d'**Euler** consiste à approximer, pour un pas de temps  $\delta t$  suffisamment petit,

$$\omega_{ij}(t + \delta t) \approx \omega_{ij}(t) + \delta t \eta [S(i, j) - \tau \omega_{ij}(t)],$$

ce qui rapproche cette méthode de la version discrète étudiée en section 2.4.1.1. Des méthodes plus sophistiquées, telles que les schémas de **Runge–Kutta** d'ordre 2 ou 4, offrent une meilleure stabilité numérique et une précision accrue en évaluant l'opérateur **F** en plusieurs points intermédiaires.

Par ailleurs, une technique d'analyse qualitative repose sur la **linéarisation** du système autour d'un **point fixe**  $\Omega^*$ . Si l'on note la perturbation  $\delta\Omega(t) = \Omega(t) - \Omega^*$ , on a

$$\frac{d}{dt} \delta\Omega(t) = D\mathbf{F}(\Omega^*) \delta\Omega(t),$$

où  $D\mathbf{F}(\Omega^*)$  est la **matrice Jacobienne** évaluée au point fixe. L'analyse des valeurs propres de cette Jacobienne permet de déterminer la **stabilité locale** : si toutes les valeurs propres ont une partie réelle négative,  $\Omega^*$  est un attracteur local stable ; sinon, des comportements tels que des cycles ou du pseudo-chaos peuvent apparaître.

#### *D. Bornage et évitement d'explosions*

Il est fréquent, pour éviter la divergence des solutions, d'imposer une condition de **bornitude** sur les pondérations, par exemple en contraignant

$$\omega_{ij}(t) \in [0, \omega_{\max}],$$

ce qui fait vivre  $\Omega(t)$  dans un ensemble  $\mathcal{X} \subset \mathbb{R}^D$  **compact**. Si l'opérateur **F** est continu et renvoie  $\mathcal{X}$  dans  $\mathcal{X}$ , on obtient alors une existence globale de la solution, et les théorèmes de point fixe (tels que le théorème de Brouwer ou celui de Schauder) assurent l'existence d'au moins un attracteur dans l'espace des pondérations. Même en l'absence de bornes strictes, le terme de décroissance  $-\tau \omega_{ij}(t)$  joue un rôle régulateur, empêchant une croissance incontrôlée des pondérations, à condition que la synergie  $S(i, j)$  ne soit pas pathologiquement infinie.

#### *E. Conclusion et synthèse*

En résumé, la dynamique continue du DSL est gouvernée par l'équation

$$\frac{d \omega_{ij}}{dt} = \eta [S(i, j; \{\omega_{p,q}\}) - \tau \omega_{ij}(t)],$$

qui, dans le cas le plus simple où  $S(i, j)$  est constant, admet une solution fermée donnée par

$$\omega_{ij}(t) = \frac{S(i, j)}{\tau} + \left( \omega_{ij}(0) - \frac{S(i, j)}{\tau} \right) \exp(-\eta \tau t).$$

Cette solution montre une convergence exponentielle vers le point fixe

$$\omega_{ij}^* = \frac{S(i, j)}{\tau}.$$

Cependant, lorsque  $S(i, j)$  dépend des pondérations elles-mêmes ou d'autres états internes, le système devient non linéaire et couplé. Dans ce cas, bien que l'existence et l'unicité de la solution soient garanties par la condition de **local Lipschitz** sur **F** (selon le théorème de Cauchy–Lipschitz), il n'existe généralement pas de solution fermée et l'analyse repose sur des méthodes numériques

telles que les schémas d'Euler ou de Runge–Kutta, ainsi que sur des techniques de linéarisation pour étudier la stabilité locale via la matrice Jacobienne  $D\mathbf{F}(\mathbf{\Omega}^*)$ .

En outre, l'introduction de mécanismes de **bornage** (par exemple, le clipping des  $\omega_{ij}$  dans l'intervalle  $[0, \omega_{\max}]$ ) assure que l'espace d'état  $\mathcal{X}$  est compact, ce qui, en conjonction avec la continuité de  $F$ , garantit l'existence globale d'au moins un attracteur. Ainsi, même si la majorité des cas non linéaires du DSL ne permettent pas une intégration fermée, la solution est en général bien définie et ne diverge pas, ce qui constitue une propriété essentielle pour l'auto-organisation du réseau.

Pour résumer, la **solution fermée** est accessible uniquement dans des situations simplifiées (par exemple, lorsque  $S(i, j)$  est constant ou linéaire et stationnaire), tandis que pour les scénarios non linéaires plus réalistes, on s'appuie sur des **approches numériques** et des techniques d'analyse qualitative. L'existence et l'unicité des solutions reposent sur la continuité et la condition de local Lipschitz de la fonction  $\mathbf{F}$ , et la présence d'un terme de décroissance  $-\tau \omega_{ij}(t)$  (ou l'application de bornes explicites) assure que la dynamique reste contractante et qu'un attracteur est présent dans l'espace des pondérations. Ces arguments illustrent comment la théorie des EDO s'applique au DSL et fournit les fondations mathématiques nécessaires à l'étude de ses propriétés dynamiques.

#### 2.4.2. Parallèles avec les Modèles de Spin (Ising, Potts, Hopfield)

Les **modèles de spin** (Ising, Potts, Hopfield) en physique statistique et en neurosciences computationnelles partagent avec le **DSL** (Deep Synergy Learning) une approche où des **entités** interagissent via des **couplages** susceptibles de se renforcer ou de conduire à des configurations d'énergie plus ou moins stable. La section 2.4.2 met en évidence ces similitudes, soulignant à la fois l'inspiration possible pour le DSL et les limites de l'analogie. Nous commençons (2.4.2.1) par la notion de **minima énergétiques** et de **transitions de phase** dans un réseau, avant d'évoquer (2.4.2.2) la comparaison  $\omega_{ij} \leftrightarrow$  couplage spin/neuron, et de modérer (2.4.2.3) la portée de ce parallèle du fait de la **non-linéarité** évolutive de la synergie en DSL.

##### 2.4.2.1. Notion de minima énergétiques, transitions de phase dans un réseau

La modélisation des systèmes complexes, qu'ils soient issus des modèles de spins classiques ou des réseaux neuronaux auto-organisés tels que le **Deep Synergy Learning (DSL)**, s'appuie souvent sur l'idée d'un paysage d'**énergie** comportant divers **minima**. Dans les modèles de spins, comme celui d'**Ising**, de **Potts** ou le réseau de **Hopfield**, la fonction d'énergie (ou **Hamiltonien**) définit une configuration stable ou un attracteur du système. Par exemple, dans le modèle d'Ising, chaque spin  $s_i$  (avec  $s_i \in \{-1, +1\}$ ) est associé à une énergie donnée par

$$\mathcal{H}(\{s_i\}) = - \sum_{\langle i, j \rangle} J_{ij} s_i s_j - \sum_i h_i s_i,$$

où la somme  $\langle i, j \rangle$  se fait sur les paires de spins voisins et  $J_{ij}$  désigne le **couplage** entre les spins  $i$  et  $j$ . Dans le modèle de **Potts**, qui généralise le modèle d'Ising à  $q$  états, le principe reste le même

: l'alignement des spins sur une valeur commune est favorisé, conduisant à des configurations d'énergie minimale. Le réseau de **Hopfield**, quant à lui, définit sa fonction d'énergie par

$$\mathcal{H} = -\frac{1}{2} \sum_{i,j} W_{ij} s_i s_j,$$

où  $W_{ij}$  représentent les poids ou **couplages synaptiques**, et les minima de cette fonction correspondent aux états mémorisés ou aux attracteurs stables du système.

Dans le cadre du DSL, bien que la dynamique ne soit pas explicitement définie en termes d'un Hamiltonien classique, l'évolution des **pondérations**  $\omega_{ij}$  peut être interprétée comme la recherche d'un état d'**optimisation** du réseau. Par analogie, on peut définir une pseudo-énergie associée aux pondérations par

$$\mathcal{E}(\Omega) \approx - \sum_{(i,j)} \omega_{ij} S(i,j),$$

où  $S(i,j)$  représente la **synergie** entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . Dans cette optique, renforcer une liaison  $\omega_{ij}$  lorsque la synergie  $S(i,j)$  est élevée équivaut à abaisser l'énergie du système, ce qui est favorable du point de vue de l'**auto-organisation**. Les **minima énergétiques** ainsi définis correspondent à des configurations stables où l'ensemble des pondérations converge vers un état attracteur, semblable aux minima locaux ou globaux observés dans les modèles de spins.

Un aspect crucial dans l'étude de ces systèmes est l'apparition de **transitions de phase**. Dans le modèle d'Ising, par exemple, on observe une transition d'un état désordonné (où les spins sont aléatoirement orientés) à un état ordonné (où la majorité des spins s'alignent) lorsque la température  $T$  chute en dessous d'un certain seuil critique  $T_c$ . Par analogie, dans un SCN, si l'on fait varier un paramètre équivalent à la température – par exemple, le niveau de bruit injecté dans la dynamique ou le rapport  $\eta/\tau$  – le système peut passer d'un régime où les pondérations sont trop fluctuantes pour permettre une structure stable à un régime où des **clusters** bien définis émergent. Cette transition, caractérisée par une « brisure de symétrie » dans l'organisation des pondérations, est analogue à la transition ordre-désordre des modèles de spins.

Les exemples de **minima énergétiques** se retrouvent, dans un DSL simplifié, lorsque l'on impose une synergie binaire ou continue mais stationnaire, ce qui permet de résoudre analytiquement l'équation de mise à jour continue (comme vu en section 2.4.1.2) et d'obtenir le point fixe

$$\omega_{ij}^* = \frac{S(i,j)}{\tau}.$$

Dans un cadre plus complexe, où  $S(i,j)$  dépend de manière non linéaire des pondérations ou d'états internes, le système présente des **attracteurs multiples** et le paysage d'énergie devient riche en minima locaux. Chaque minimum représente une configuration stable locale du réseau, correspondant à un cluster ou à une partition particulière de l'ensemble des entités. Les **transitions de phase** se manifestent alors par des changements brusques dans l'organisation du réseau lorsqu'un paramètre (tel que le bruit ou le rapport  $\eta/\tau$ ) atteint un seuil critique, conduisant à un basculement d'un attracteur à un autre.

En résumé, les modèles de spins – Ising, Potts, Hopfield – nous fournissent un vocabulaire riche qui peut être transposé au DSL, où l’on parle de **minima énergétiques** et de **transitions de phase** pour interpréter les configurations stables des pondérations. Un **minimum local** d’énergie dans le DSL correspond à une configuration des poids  $\Omega$  qui maximise la synergie interne, et la transition d’un état diffus à un état clusterisé s’interprète comme une transition ordre-désordre. Bien que la synergie  $S(i, j)$  dans le DSL puisse varier dans le temps ou dépendre d’états internes, l’analogie avec la minimisation d’un Hamiltonien offre une perspective éclairante pour comprendre l’émergence des **clusters** et l’existence d’attracteurs dans l’espace des pondérations.

## Conclusion

La notion de **minima énergétiques** et de **transitions de phase** dans les modèles de spins trouve un parallèle évident dans l’analyse du DSL. Dans les deux cadres, l’optimisation d’une fonction (qu’elle soit l’énergie ou une pseudo-énergie définie par  $\mathcal{E}(\Omega) \approx -\sum_{(i,j)} \omega_{ij} S(i, j)$ ) conduit à la formation d’états stables, et des transitions abruptes peuvent apparaître lorsque des paramètres critiques sont franchis. Ces analogies, en reliant les couplages  $\omega_{ij}$  aux interactions de spins et en interprétant la dynamique des pondérations comme une recherche de minima dans un paysage d’énergie, fournissent un langage commun permettant de comprendre les phénomènes d’auto-organisation observés dans le DSL, même lorsque la fonction de synergie varie dans le temps ou dépend de plusieurs facteurs. Les sections suivantes approfondiront ces correspondances et examineront les limites de cette analogie, notamment lorsque la dynamique du DSL intègre des aspects temporels et non linéaires complexes.

### 2.4.2.2. Interprétation de $\omega_{ij}$ comme “couplage” entre spins ou neurones

Dans les modèles classiques de la physique statistique, tels que les modèles de **spins d’Ising** ou de **Potts**, ainsi que dans les réseaux de **Hopfield**, la notion de couplage est centrale. Dans le modèle d’Ising, par exemple, chaque spin  $s_i$  prenant des valeurs dans  $\{-1, +1\}$  interagit avec ses voisins via des paramètres  $J_{ij}$  qui, selon leur signe, favorisent l’alignement ou l’antialignement. La fonction d’énergie (Hamiltonien) s’exprime typiquement sous la forme

$$\mathcal{H}(\{s_i\}) = -\sum_{\langle i,j \rangle} J_{ij} s_i s_j - \sum_i h_i s_i,$$

où la somme  $\langle i, j \rangle$  parcourt les paires de spins voisins et  $h_i$  représente un champ externe appliqué sur le site  $i$ . Dans le modèle de Potts, la généralisation aux  $q$  états se traduit par une évaluation similaire de la similitude ou de la différence d’états entre spins, tandis que dans le réseau de Hopfield, l’énergie est définie par

$$E(\mathbf{s}) = -\frac{1}{2} \sum_{i,j} W_{ij} s_i s_j,$$

les poids  $W_{ij}$  constituant alors les **couplages synaptiques** qui restent fixés au cours du temps et dont les minima énergétiques correspondent aux états stables du système (mémoires associatives).

Dans le contexte du **Deep Synergy Learning (DSL)**, la pondération  $\omega_{ij}$  remplit une fonction analogue à celle des couplages  $J_{ij}$  ou  $W_{ij}$  dans les modèles de spins. En effet,  $\omega_{ij}$  quantifie la force du lien entre deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . Une valeur élevée de  $\omega_{ij}$  signifie que la synergie entre les entités, mesurée par  $S(i, j)$ , est forte et que le lien entre elles est donc très robuste. La dynamique du DSL repose sur une mise à jour de type

$$\omega_{ij}(t + 1) = \omega_{ij}(t) + \eta [S(i, j) - \tau \omega_{ij}(t)],$$

ce qui signifie que la pondération évolue en fonction de la différence entre la synergie effective et un terme de décroissance proportionnel à  $\omega_{ij}$ . À cet égard,  $\omega_{ij}$  joue un rôle similaire à un couplage adaptatif. En effet, dans les réseaux de Hopfield ou d'Ising, le couplage entre deux spins est un paramètre constant qui oriente l'alignement des états  $s_i$  et  $s_j$ . Dans le DSL, ce sont les **liaisons** elles-mêmes qui évoluent en temps réel en réponse aux mesures de synergie, de sorte que  $\omega_{ij}$  est non seulement un indicateur de la force de connexion, mais également un paramètre qui se met à jour pour refléter l'état d'interaction actuel entre les entités.

Une autre différence fondamentale réside dans le fait que, dans les modèles classiques, l'énergie du système est exprimée directement en fonction des spins, tandis que dans le DSL, l'analogie se construit en définissant une pseudo-énergie du réseau. Par exemple, on peut envisager la fonction

$$\mathcal{E}(\Omega) \approx - \sum_{(i,j)} \omega_{ij} S(i, j),$$

qui traduit l'idée que renforcer les pondérations  $\omega_{ij}$  lorsque  $S(i, j)$  est élevée réduit l'énergie globale du système, à la manière dont un couplage  $J_{ij}$  élevé favorise un état ordonné dans un modèle de spins. Cette interprétation permet d'assimiler, d'une certaine manière, le rôle de  $\omega_{ij}$  à celui d'un couplage classique, même si la synergie  $S(i, j)$  dans le DSL peut résulter de calculs plus complexes qu'un simple produit scalaire ou une simple corrélation.

Par ailleurs, la dynamique DSL se distingue par le fait que, contrairement aux modèles d'Ising ou de Hopfield où les couplages sont fixes et ce sont les états (spins ou activations) qui évoluent, dans le DSL, ce sont les **liaisons** elles-mêmes qui évoluent tandis que l'état interne de chaque entité peut rester relativement stable ou évoluer indépendamment. Cette inversion dans le rôle des variables signifie que le DSL est axé sur l'adaptation des **relations** entre les entités, ce qui permet au réseau de se réorganiser de manière flexible en réponse à de nouvelles informations.

Enfin, il convient d'envisager les extensions possibles de cette analogie vers des interactions plus complexes. Tandis que les modèles classiques de spins se limitent à des couplages binaires, le DSL ouvre la possibilité d'introduire des interactions **n-aires**. Ces interactions, qui impliquent la synergie collective de plusieurs entités, peuvent être représentées dans un cadre d'**hypergraphes** où les hyper-liens traduisent des relations qui ne se décomposent pas simplement en interactions par paires. Même si cette généralisation dépasse le cadre des modèles de spins classiques, elle enrichit la compréhension du DSL en offrant une perspective plus large sur la **coopération** entre entités.

En conclusion, l'interprétation de  $\omega_{ij}$  dans le DSL comme un **couplage** entre entités s'appuie sur plusieurs points de convergence avec les modèles de spins et les réseaux de Hopfield. D'une part,

$\omega_{ij}$  joue le rôle de modulateur de la force d'interaction, tout comme  $J_{ij}$  ou  $W_{ij}$  favorisent l'alignement des spins dans les modèles classiques. D'autre part, la dynamique évolutive des pondérations, qui suit une mise à jour continue ou discrète, permet au DSL de s'adapter de manière dynamique, en modifiant les liaisons en fonction de la **synergie effective** mesurée entre les entités. Bien que cette dynamique diffère du cadre statique traditionnel des modèles de spins, l'analogie reste pertinente et fournit un langage commun pour l'analyse des attracteurs, des transitions de phase et de l'auto-organisation dans les réseaux complexes. Les développements ultérieurs aborderont les limites de cette analogie et exploreront comment l'intégration de synergies n-aires peut enrichir cette approche en modélisant des interactions plus collectives et sophistiquées.

### 2.4.2.3. Limites de la comparaison : le DSL introduit une synergie “non-linéaire” pouvant varier au fil du temps

Dans les sections précédentes, les analogies établies entre le **DSL** (Deep Synergy Learning) et les modèles de spin classiques – notamment ceux d’Ising, de Potts ou de Hopfield – se fondent sur l’assimilation de la pondération  $\omega_{ij}$  à un **couplage** comparable aux paramètres  $J_{ij}$  ou  $W_{ij}$ . Dans ces modèles, le couplage est généralement considéré comme une constante fixe, tandis que dans le DSL, la dynamique des pondérations se fait évoluer en temps réel. Toutefois, cette comparaison présente des limites majeures, car la **synergie**  $S(i, j)$  dans le DSL est souvent caractérisée par une nature **non-linéaire** et par une variabilité temporelle qui ne se retrouve pas dans les modèles de spins stationnaires.

Dans les modèles classiques tels que ceux d’Ising ou de Potts, chaque spin  $s_i \in \{-1, +1\}$  interagit avec ses voisins via des couplages  $J_{ij}$  qui sont fixés dans le temps, sauf dans certaines variantes où l’on peut envisager une plasticité synaptique. Par exemple, l’énergie du modèle d’Ising est donnée par

$$\mathcal{H}(\{s_i\}) = - \sum_{\langle i, j \rangle} J_{ij} s_i s_j - \sum_i h_i s_i,$$

où la matrice  $\{J_{ij}\}$  est supposée constante, permettant ainsi une descente d’énergie par l’évolution des spins  $\{s_i\}$ . Dans un réseau de Hopfield, les poids  $\{W_{ij}\}$  sont également fixés après apprentissage et la dynamique repose sur l’évolution des activations  $s_i$  pour minimiser la fonction d’énergie

$$\mathcal{H}(\mathbf{s}) = - \frac{1}{2} \sum_{i, j} W_{ij} s_i s_j.$$

En contraste, dans le DSL, la pondération  $\omega_{ij}$  évolue en fonction de la synergie mesurée, par exemple selon une règle de mise à jour discrète

$$\omega_{ij}(t + 1) = \omega_{ij}(t) + \eta [S(i, j) - \tau \omega_{ij}(t)],$$

où  $\eta$  est le **taux d’apprentissage** et  $\tau$  le **coefficient de décroissance**. Ici, la synergie  $S(i, j)$  peut être issue de diverses mesures (distance, similarité, co-information) et, contrairement aux couplages classiques, elle peut dépendre de variables évolutives telles que les représentations  $\mathbf{x}_i$  des entités ou même des états internes  $\mathbf{s}_i$ . De plus, dans certaines variantes du DSL,  $S(i, j)$  intègre des termes dépendant d’autres pondérations, par exemple sous la forme d’un terme d’**inhibition compétitive** tel que



$$S(i, j) \propto -\gamma \sum_{k \neq j} \omega_{i,k},$$

ce qui introduit une non-linéarité supplémentaire dans la dynamique. Par conséquent, la fonction  $S(i, j)$  ne se réduit pas à un simple produit scalaire  $s_i \times s_j$  comme dans les modèles de spins, mais peut varier en fonction des **représentations** et évoluer en temps réel.

Cette variabilité pose des problèmes pour établir une analogie rigoureuse avec une fonction d'énergie stationnaire. En effet, dans un système tel que celui de Hopfield, l'énergie

$$\mathcal{H}(\mathbf{s}) = -\frac{1}{2} \sum_{i,j} W_{ij} s_i s_j$$

reste fixe en fonction des couplages  $W_{ij}$  tandis que les spins évoluent. Dans le DSL, en revanche, les pondérations  $\omega_{ij}$  ainsi que la synergie  $S(i, j)$  se réévaluent continuellement, ce qui rend difficile l'établissement d'un paysage d'énergie immuable, c'est-à-dire d'une fonction  $\mathcal{E}(\boldsymbol{\Omega})$  telle que

$$\mathcal{E}(\boldsymbol{\Omega}) \approx - \sum_{(i,j)} \omega_{ij} S(i, j)$$

soit une mesure stable sur laquelle le système effectuerait une descente d'énergie. Par ailleurs, la dynamique du DSL se caractérise par le fait que les entités elles-mêmes, représentées par leurs vecteurs  $\mathbf{x}_i$  ou leurs états internes  $\mathbf{s}_i$ , sont relativement stables par rapport aux pondérations. Ainsi, dans les modèles classiques, ce sont les spins qui varient tandis que le couplage reste fixe, tandis que dans le DSL, le couplage  $\omega_{ij}$  évolue, ce qui inverse la dynamique habituelle.

De plus, la synergie  $S(i, j)$  peut être influencée par des mécanismes complexes et non linéaires, tels que la dépendance aux états internes ou l'intégration d'une co-information multi-entités. Par conséquent, le paysage d'énergie du DSL est en constante évolution, et une configuration de pondérations optimale à un instant donné peut se transformer à l'instant suivant si la synergie se modifie. Ce caractère dynamique et non stationnaire limite la validité d'une comparaison stricte avec des modèles de spins classiques, où l'on dispose d'un Hamiltonien fixe et d'une descente d'énergie bien définie.

En conclusion, bien que l'analogie entre  $\omega_{ij}$  dans le DSL et les couplages  $J_{ij}$  ou  $W_{ij}$  des modèles de spins fournisse un langage utile pour comprendre la formation des clusters et la multi-stabilité du système, elle reste partielle. Le DSL introduit une **synergie non-linéaire** et variable dans le temps, ce qui complexifie la notion d'un paysage d'énergie statique et remet en question l'application directe des concepts de descente d'énergie et de transitions de phase issus de la physique statistique. Néanmoins, cette analogie demeure précieuse pour interpréter qualitativement les phénomènes d'**attraction** et de **basculement** observés dans le DSL, tout en reconnaissant que la dynamique de ce système dépasse largement la structure d'un couplage fixe et d'un Hamiltonien stationnaire.

### *Conclusion*

La comparaison entre le DSL et les modèles de spins, bien que riche en enseignements, présente des limites importantes. Dans les modèles classiques, le couplage est fixe et l'énergie est définie de

manière stationnaire, ce qui permet une analyse rigoureuse de la descente d'énergie et des transitions de phase. En revanche, dans le DSL, la **synergie**  $S(i, j)$  est souvent non linéaire et évolutive, ce qui empêche de définir un paysage d'énergie immuable et complique l'interprétation quantitative des attracteurs et des transitions de phase. Cette analogie, bien qu'instructive, ne peut être considérée que comme partielle et doit être nuancée en tenant compte de la nature dynamique et variable du DSL, qui intègre des mécanismes d'adaptation et des interactions évolutives dépassant le cadre d'un couplage statique.

### 2.4.3. Notion d'Énergie ou de Fonction Potentielle

Les **modèles de spin** (Ising, Potts, Hopfield) s'appuient sur une **énergie**  $\mathcal{H}$  (ou  $\mathcal{E}$ ) que l'évolution du système vise à **minimiser** ou dont il cherche à **explorer** les minima. Dans le cadre du **Deep Synergy Learning** (DSL), on aimerait parfois définir un **analogue** de cette fonction d'énergie ou de potentiel, afin de mieux comprendre la **stabilité** et la **convergence** du SCN (Synergistic Connection Network). La section 2.4.3 soulève la question de savoir si on peut formaliser  $\mathcal{J}(\Omega)$  (2.4.3.1), examiner l'existence de "descente d'énergie" (2.4.3.2) et évaluer les avantages/inconvénients de cette approche (2.4.3.3).

#### 2.4.3.1. Peut-on formaliser $\mathcal{J}(\Omega) = -\sum_{i,j} \omega_{i,j} S(i, j) + \dots$ ?

Dans l'étude du **Deep Synergy Learning** (DSL), une question centrale consiste à savoir si la dynamique des pondérations  $\omega_{ij}$  peut être interprétée comme une descente d'énergie, analogue à la minimisation d'un Hamiltonien dans les modèles de spins. Pour ce faire, on cherche à définir une fonction de coût ou une **fonction énergétique**  $\mathcal{J}(\Omega)$  dont la minimisation conduirait naturellement aux mises à jour observées dans le DSL. Une proposition heuristique couramment avancée est de poser

$$\mathcal{J}(\Omega) = -\sum_{i,j} \omega_{ij} S(i, j) + (\text{termes de régularisation}),$$

où  $S(i, j)$  désigne la **synergie effective** entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . Le premier terme,  $-\omega_{ij} S(i, j)$ , est conçu pour encourager l'augmentation des pondérations lorsque la synergie est élevée, c'est-à-dire qu'en minimisant  $\mathcal{J}$  on favorise une configuration dans laquelle les liens forts correspondent aux synergies importantes. Pour éviter que les pondérations ne croissent indéfiniment, on ajoute généralement un terme de régularisation qui peut prendre la forme d'une pénalisation quadratique, par exemple

$$\frac{\tau}{2} \sum_{i,j} \omega_{ij}^2,$$

de sorte que l'expression complète devient

$$\mathcal{J}(\Omega) = -\sum_{i,j} \omega_{ij} S(i, j) + \frac{\tau}{2} \sum_{i,j} \omega_{ij}^2 + \dots,$$

où  $\tau > 0$  agit comme un **paramètre de régularisation**. Le gradient partiel par rapport à une pondération  $\omega_{ij}$  s'obtient alors en différentiant  $\mathcal{J}$  par rapport à  $\omega_{ij}$ , ce qui conduit à

$$\frac{\partial \mathcal{J}}{\partial \omega_{ij}} = -S(i, j) + \tau \omega_{ij}.$$

Ainsi, si l'on interprète la règle de mise à jour du DSL comme un pas de gradient négatif, on obtient

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \eta \frac{\partial \mathcal{J}}{\partial \omega_{ij}} \big|_{\omega_{ij}(t)} = \omega_{ij}(t) + \eta [S(i, j) - \tau \omega_{ij}(t)],$$

ce qui correspond exactement à la mise à jour standard utilisée dans le DSL dans le cas stationnaire où  $S(i, j)$  est indépendant de  $\omega_{ij}$  et des autres pondérations.

Cette correspondance formelle permet de comprendre que, dans un cadre idéal et simplifié, le DSL opère comme une **descente de gradient** sur la fonction  $\mathcal{J}$ , ce qui aurait pour effet de minimiser l'énergie globale du réseau. L'hypothèse sous-jacente est que la synergie  $S(i, j)$  est stationnaire ou du moins qu'elle varie lentement par rapport aux mises à jour des pondérations. Dans ce cas, le paysage énergétique défini par

$$\mathcal{J}(\boldsymbol{\Omega}) = - \sum_{i,j} \omega_{ij} S(i, j) + \frac{\tau}{2} \sum_{i,j} \omega_{ij}^2$$

possède un minimum global que le système cherche à atteindre, avec pour point fixe

$$\omega_{ij}^* = \frac{S(i, j)}{\tau},$$

ce qui reflète une convergence exponentielle, comme nous l'avons vu dans l'analyse de la variante continue.

Cependant, des **obstacles** apparaissent dès lors que l'on introduit des éléments de non-linéarité dans la fonction de synergie. Lorsque  $S(i, j)$  dépend de manière non triviale des autres pondérations  $\{\omega_{p,q}\}$  ou d'états internes variables, le terme  $S(i, j)$  devient lui-même une fonction de  $\boldsymbol{\Omega}$  et ne peut plus être considéré comme une constante. Dans ce cas, la dérivation de  $\mathcal{J}$  ne se réduit plus à une simple fonction linéaire en  $\omega_{ij}$  et l'opérateur de mise à jour

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta [S(i, j; \{\omega_{p,q}(t)\}) - \tau \omega_{ij}(t)]$$

ne correspond plus à une descente de gradient d'une unique fonction énergétique  $\mathcal{J}(\boldsymbol{\Omega})$  fixe. Le paysage énergétique devient alors **dynamique** et se reconfigure en continu, reflétant des interactions complexes et souvent non stationnaires. En d'autres termes, la fonction  $S(i, j)$  peut évoluer en réponse aux changements dans  $\boldsymbol{\Omega}$  et aux entrées extérieures, ce qui empêche la formulation d'un Hamiltonien fixe et stable comme dans les modèles de spins classiques.

Dans cette situation, il est souvent nécessaire de recourir à des **approches numériques** pour étudier l'évolution du système et pour déterminer qualitativement l'existence et la nature des attracteurs. On peut ainsi utiliser des schémas d'intégration numérique (comme Euler ou Runge–Kutta) pour approximer la trajectoire  $\omega_{ij}(t)$ , et des techniques de **linéarisation** autour des points fixes pour

analyser la stabilité locale à l'aide de la matrice Jacobienne  $\mathbf{DF}(\mathbf{\Omega}^*)$ . Ces méthodes permettent de caractériser les attracteurs locaux, même si l'on ne peut pas écrire explicitement une solution fermée pour l'ensemble des pondérations.

De plus, des contraintes de **bornage** (par exemple, imposer que  $\omega_{ij} \in [0, \omega_{\max}]$ ) ou l'utilisation de mécanismes de **saturation** garantissent que l'espace des solutions reste compact, ce qui est indispensable pour appliquer les théorèmes classiques de point fixe (Brouwer, Schauder). Cela permet d'assurer, malgré l'absence de solution analytique globale, que le système admet au moins un attracteur dans l'espace des configurations, même si ce dernier dépend de paramètres externes et de la dynamique non linéaire de  $S(i, j)$ .

En synthèse, la formalisation d'une fonction énergétique globale dans le DSL se fait aisément dans des cas **simples** où  $S(i, j)$  est constant ou linéaire, ce qui conduit à une solution fermée du type

$$\omega_{ij}(t) = \frac{S(i, j)}{\tau} + \left( \omega_{ij}(0) - \frac{S(i, j)}{\tau} \right) \exp(-\eta \tau t).$$

Dans les scénarios plus **complexes** et non linéaires, la dynamique du DSL ne correspond pas à la descente d'un Hamiltonien fixe, mais plutôt à un processus adaptatif évoluant dans un paysage énergétique variable. Les approches pour étudier ces systèmes reposent alors sur des méthodes numériques et sur l'analyse qualitative via la linéarisation locale. Ainsi, bien que la vision «  $\mathcal{J}(\mathbf{\Omega}) = -\sum_{i,j} \omega_{ij} S(i, j) + \frac{\tau}{2} \sum_{i,j} \omega_{ij}^2 + \dots$  » soit pertinente dans un cadre stationnaire, elle doit être nuancée lorsque la synergie dépend de l'état du système et varie dans le temps.

### Conclusion

On peut formaliser, dans un scénario restreint, une fonction énergétique

$$\mathcal{J}(\mathbf{\Omega}) = -\sum_{i,j} \omega_{ij} S(i, j) + \frac{\tau}{2} \sum_{i,j} \omega_{ij}^2 + \dots,$$

telle que la mise à jour des pondérations s'apparente à une descente de gradient

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \eta \frac{\partial \mathcal{J}}{\partial \omega_{ij}} \big|_{\omega_{ij}(t)} = \omega_{ij}(t) + \eta [S(i, j) - \tau \omega_{ij}(t)].$$

Cependant, dès que la fonction  $S(i, j)$  présente des dépendances non linéaires ou évolutives en fonction de  $\mathbf{\Omega}$  ou d'états internes, la notion d'un paysage d'énergie unique et fixe devient moins pertinente, et le système évolue dans un cadre « hors équilibre ». Dans ce cas, l'approche repose sur des méthodes numériques et une analyse qualitative de la dynamique. Ainsi, la question d'une solution fermée se résout par un “oui” dans des cas idéalisés et par un “non” dans la majorité des situations réalistes, nécessitant des approches approximatives pour garantir l'existence et la stabilité des solutions du DSL.

### 2.4.3.2. Existence de descentes d'énergie locales : recuit simulé ou itération stochastique

Dans le cadre du **Deep Synergy Learning (DSL)**, l'idée d'interpréter la dynamique des pondérations  $\omega_{ij}$  comme une descente dans un paysage énergétique a suscité un intérêt particulier, en raison de ses analogies avec les méthodes d'optimisation stochastique telles que le **recuit simulé** (simulated annealing). Nous allons examiner ici la possibilité de formaliser une **descente d'énergie locale** dans le DSL, ainsi que les méthodes par lesquelles un tel processus, éventuellement perturbé par du bruit aléatoire, peut permettre au système d'explorer différents minima énergétiques et ainsi d'échapper à des minima locaux non optimaux.

Dans les modèles de spins classiques, par exemple dans le modèle d'**Ising** ou dans les réseaux de **Hopfield**, l'algorithme de recuit simulé est utilisé pour échapper aux pièges des minima locaux. Le principe est d'introduire une **température**  $T$  qui module la probabilité d'accepter une augmentation de l'énergie  $\Delta\mathcal{H}$  selon une loi de probabilité de type

$$P(\Delta\mathcal{H}) \propto \exp\left(-\frac{\Delta\mathcal{H}}{T}\right).$$

Lorsque  $T$  est élevé, le système explore librement l'espace des configurations, même si certaines transitions font temporairement augmenter l'énergie. Au fil du temps,  $T$  diminue progressivement, limitant ainsi les fluctuations et permettant au système de converger vers un minimum global ou, du moins, vers un minimum local satisfaisant.

Pour transposer cette approche au DSL, on suppose qu'il est possible de définir une **pseudo-énergie**  $\mathcal{J}(\mathbf{\Omega})$  qui dépend de l'ensemble des pondérations  $\omega_{ij}$  et qui prend la forme

$$\mathcal{J}(\mathbf{\Omega}) = - \sum_{i,j} \omega_{ij} S(i,j) + \frac{\tau}{2} \sum_{i,j} \omega_{ij}^2 + \dots,$$

où  $S(i,j)$  représente la **synergie effective** entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . Dans le cas stationnaire et en l'absence de dépendances complexes de  $S$  en fonction de  $\mathbf{\Omega}$ , la descente de gradient appliquée à  $\mathcal{J}$  conduit à une mise à jour

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \eta \frac{\partial \mathcal{J}}{\partial \omega_{ij}} \Big|_{\omega_{ij}(t)} = \omega_{ij}(t) + \eta [S(i,j) - \tau \omega_{ij}(t)],$$

ce qui correspond exactement à la règle de mise à jour usuelle du DSL. Ce constat démontre que, dans des cas idéalisés, le DSL effectue effectivement une **descente d'énergie locale** vers le point fixe  $\omega_{ij}^* = S(i,j)/\tau$ .

Néanmoins, la réalité est souvent plus complexe. Dans de nombreuses applications, la fonction de synergie  $S(i,j)$  est sujette à des **non-linéarités** et peut varier en fonction des autres pondérations  $\{\omega_{p,q}\}$  ou des états internes des entités. Pour intégrer ce comportement, on peut modifier la règle de mise à jour en y ajoutant un terme de bruit stochastique, de sorte que l'équation prenne la forme

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \eta \frac{\partial \mathcal{J}}{\partial \omega_{ij}} \Big|_{\omega(t)} + \sigma \xi_{ij}(t),$$

où  $\xi_{ij}(t)$  représente une perturbation aléatoire et  $\sigma$  en détermine l'amplitude. Ce terme de bruit permet d'imaginer une dynamique d'**itération stochastique** ou de **recuit simulé** qui, en phase initiale, autorise le système à franchir des barrières énergétiques, explorant ainsi différents minima du paysage. Au fil du temps, un **planning de refroidissement** (par exemple, une décroissance exponentielle de  $\sigma$  telle que  $\sigma(t) \propto \alpha^t$  avec  $0 < \alpha < 1$ ) est appliqué afin de réduire progressivement l'influence du bruit, permettant alors au système de se stabiliser dans une configuration potentiellement plus globale.

La démarche proposée s'inspire directement des algorithmes de **recuit simulé** utilisés dans les modèles d'optimisation stochastique. En Ising ou dans une machine de Boltzmann, on utilise des critères probabilistes pour accepter ou refuser des changements qui augmentent l'énergie, ce qui permet de surmonter le problème des minima locaux. Dans le DSL, bien que le paysage énergétique ne soit pas nécessairement fixe – en particulier lorsque la synergie  $S(i, j)$  est fonction des pondérations elles-mêmes – l'injection d'un bruit contrôlé offre un levier pour encourager l'exploration et éviter que le réseau ne se bloque prématurément dans une configuration sous-optimale.

Pour illustrer cette approche, il convient d'imaginer un protocole « à la recuit simulé » appliqué aux pondérations. Dans la **phase chaude**, le niveau de bruit  $\sigma(0)$  est élevé, ce qui autorise de fortes fluctuations dans  $\omega_{ij}$  et favorise l'exploration de configurations variées. Puis, lors d'un **refroidissement progressif**,  $\sigma(t)$  décroît de manière prédéfinie, par exemple suivant une loi exponentielle ou linéaire, et par conséquent, les modifications aléatoires deviennent de plus en plus rares. Finalement, dans la **phase finale** où  $\sigma(t) \approx 0$ , le système se stabilise dans l'un des minima locaux de la fonction pseudo-énergétique  $\mathcal{J}(\Omega)$ .

Ce mécanisme est particulièrement pertinent dans les situations où la fonction  $S(i, j)$  est suffisamment stationnaire pour que la notion de descente d'énergie ait un sens. Toutefois, dans des scénarios où  $S(i, j)$  varie de manière significative en fonction des pondérations ou des états internes, le paysage d'énergie devient mouvant et la descente n'est qu'une **approximation** de la dynamique réelle. Dans ce cas, le recuit simulé ne garantit pas la minimisation d'une fonction énergétique fixe, mais sert plutôt de stratégie pour favoriser l'exploration et pour empêcher le système de se figer dans des minima locaux non désirés.

## Conclusion

Ainsi, dans un cadre restreint où la synergie  $S(i, j)$  reste relativement stationnaire, il est possible de formaliser la dynamique du DSL comme une descente d'énergie locale à partir d'une fonction pseudo-énergétique

$$\mathcal{J}(\Omega) = - \sum_{i,j} \omega_{ij} S(i, j) + \frac{\tau}{2} \sum_{i,j} \omega_{ij}^2 + \dots,$$

ce qui conduit à la mise à jour

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \eta \frac{\partial \mathcal{J}}{\partial \omega_{ij}} \Big|_{\omega_{ij}(t)} = \omega_{ij}(t) + \eta [S(i, j) - \tau \omega_{ij}(t)].$$

Pour explorer d'autres configurations et potentiellement échapper à des minima locaux, un terme de bruit  $\sigma \xi_{ij}(t)$  peut être ajouté, reproduisant ainsi le comportement d'un recuit simulé. Ce

protocole stochastique permet au système d’explorer divers attracteurs avant de converger vers une configuration stable lorsque le niveau de bruit décroît progressivement. Cependant, si la synergie  $S(i, j)$  dépend fortement des pondérations ou d’états internes variables, le paysage d’énergie n’est plus fixe et la descente d’énergie devient une approximation qui ne capture que partiellement la dynamique réelle. Néanmoins, l’introduction d’un recuit simulé ou d’itérations stochastiques dans le DSL constitue un levier important pour encourager l’exploration et pour éviter un blocage prématuré du système dans un minimum local sous-optimum.

En conclusion, le DSL peut, dans des conditions appropriées, être interprété comme effectuant une descente d’énergie locale, et l’injection contrôlée de bruit – de type recuit simulé – permet d’explorer le paysage énergétique et d’optimiser la structure du réseau. Cette approche, bien que dépendante d’hypothèses stationnaires, fournit une base théorique et pratique pour comprendre comment le système peut échapper aux minima locaux et potentiellement converger vers une configuration plus optimale.

#### 2.4.3.3. Avantages et inconvénients de cette vision “énergétique” pour expliquer la convergence

L’approche consistant à interpréter la dynamique du DSL à travers une fonction d’énergie  $J(\Omega)$  offre une perspective particulièrement intéressante pour analyser la convergence des pondérations  $\omega_{ij}$  dans un Synergistic Connection Network (DSL). En effet, en posant par exemple

$$J(\Omega) = - \sum_{i,j} \omega_{ij} S(i, j) + \frac{\tau}{2} \sum_{i,j} \omega_{ij}^2 + \dots,$$

on traduit l’idée que le système cherche à maximiser la synergie entre entités tout en régulant la croissance des pondérations à travers un terme quadratique de **régularisation**. Dans ce cadre, la mise à jour

$$\omega_{ij}(t + 1) = \omega_{ij}(t) - \eta \frac{\partial J}{\partial \omega_{ij}} \bigg|_{\omega_{ij}(t)}$$

se simplifie en

$$\omega_{ij}(t + 1) = \omega_{ij}(t) + \eta [S(i, j) - \tau \omega_{ij}(t)],$$

ce qui correspond exactement à la règle de mise à jour adoptée dans le DSL dans un cadre stationnaire. Dans cette situation idéale, la dynamique s’apparente à une **descente de gradient** dans un paysage d’énergie fixe, permettant d’expliquer l’**auto-organisation** des pondérations comme une recherche d’un minimum local de  $J$ . Le principal avantage de cette vision est qu’elle fournit un cadre conceptuel puissant et familier aux chercheurs en physique statistique. Par analogie avec les modèles d’Ising et de Hopfield, où l’énergie  $\mathcal{H}(\mathbf{s})$  guide l’évolution des spins, cette approche énergétique permet de justifier l’émergence d’attracteurs et de **clusters** en termes de minimisation d’un potentiel global.

D’un point de vue théorique, cette formalisation simplifie l’analyse des propriétés de convergence, car elle permet de mobiliser des outils issus de l’**optimisation** et de la **théorie des systèmes**

**dynamiques.** En effet, dans le cas où la synergie  $S(i, j)$  demeure relativement stationnaire et ne dépend pas trop des pondérations elles-mêmes, le paysage d'énergie défini par  $\mathcal{J}(\Omega)$  est stable et la dynamique converge vers un minimum global ou, au moins, vers un attracteur local caractérisé par

$$\omega_{ij}^* = \frac{S(i, j)}{\tau}.$$

Cette interprétation favorise l'emploi de stratégies de **recuit simulé** ou de **gradient stochastique** qui permettent d'explorer l'espace des configurations et d'éviter de se figer prématurément dans des minima locaux sous-optimaux. Le concept de "température" dans ces méthodes, qui module l'amplitude du bruit injecté dans la mise à jour, offre un levier pour contrôler le compromis entre exploration et exploitation du paysage énergétique.

Cependant, cette vision présente également des **limites** importantes lorsqu'elle est appliquée à des scénarios réels du DSL. En effet, dans la majorité des cas, la fonction de synergie  $S(i, j)$  n'est pas une valeur fixe mais une fonction **non-linéaire** qui peut dépendre des autres pondérations  $\{\omega_{p,q}\}$ , des états internes  $\mathbf{s}_i$  et même de paramètres externes. Ainsi, le paysage d'énergie  $\mathcal{J}(\Omega)$  n'est pas immuable et se reconfigure en continu. Ce phénomène remet en cause l'hypothèse d'un **Hamiltonien stationnaire** et limite l'applicabilité de la descente d'énergie telle que décrite dans les modèles classiques. Dans un tel cadre, même si la mise à jour locale des pondérations peut être interprétée comme une descente de gradient sur une fonction d'énergie approximative, il est difficile de garantir que le système converge vers un minimum global ou même vers un attracteur stable, puisque la fonction  $S(i, j)$  elle-même évolue en temps réel.

De plus, contrairement aux modèles d'Ising ou de Hopfield, dans lesquels les couplages (comme  $J_{ij}$  ou  $W_{ij}$ ) sont généralement **fixes** et seuls les états (spins, activations) varient, le DSL fait évoluer les pondérations  $\omega_{ij}$  de manière adaptative. Cette dynamique, qui s'appuie sur des mécanismes d'auto-organisation complexes, engendre un système souvent **hors équilibre**, dans lequel la notion de descente d'un paysage énergétique fixe est plus difficile à appliquer de manière rigoureuse.

En synthèse, la **vision énergétique** du DSL offre des avantages considérables, notamment en permettant d'interpréter la formation de clusters et la convergence des pondérations comme le résultat d'une **descente de gradient** dans un paysage d'énergie. Ce cadre permet d'employer des techniques d'**optimisation stochastique** et de recuit simulé pour favoriser l'exploration de l'espace de configurations et éviter le piégeage dans des minima locaux peu optimaux. Néanmoins, lorsque la synergie  $S(i, j)$  varie significativement en fonction des pondérations et des états internes, le paysage d'énergie devient dynamique, et la correspondance avec un Hamiltonien stationnaire s'efface. Par conséquent, cette approche doit être considérée comme une **approximation** utile dans des scénarios stationnaires ou faiblement dépendants, mais elle perd en pertinence lorsque le système se comporte de manière fortement non linéaire et adaptative. Ainsi, bien que l'analogie avec une fonction d'énergie soit instructive pour expliquer qualitativement la convergence du DSL, elle présente des limites quant à sa capacité à rendre compte de la totalité de la dynamique du système.



## Conclusion

La vision énergétique du DSL, qui repose sur l'hypothèse que les pondérations  $\omega_{ij}$  évoluent de manière à minimiser une fonction pseudo-énergétique telle que

$$\mathcal{J}(\Omega) = - \sum_{i,j} \omega_{ij} S(i,j) + \frac{\tau}{2} \sum_{i,j} \omega_{ij}^2 + \dots,$$

permet de comprendre la formation de clusters et la convergence du réseau dans des cadres stationnaires. Elle offre un langage commun avec les modèles de spins classiques et autorise l'utilisation de méthodes d'optimisation stochastique telles que le recuit simulé pour éviter le piégeage dans des minima locaux. Toutefois, cette approche reste partielle car, dans la majorité des scénarios réels du DSL, la synergie  $S(i,j)$  est non linéaire et évolutive, ce qui entraîne une reconfiguration continue du paysage d'énergie et rend difficile l'application d'une descente d'un potentiel immuable. La convergence observée dans le DSL ne peut alors être expliquée que partiellement par la minimisation d'une fonction énergétique, et doit être complétée par une analyse qualitative de la dynamique adaptative du système.

### 2.4.4. Rôle de l'Inhibition et de la Saturation Synaptique

Dans un SCN (Synergistic Connection Network), la simple règle  $\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i,j) - \tau \omega_{i,j}(t)]$  peut, dans certaines conditions, conduire à un **développement excessif** de liaisons (pour peu que  $S(i,j)$  reste élevée) ou, au contraire, à l'émergence d'oscillations indésirables. Des **mécanismes** supplémentaires, tels que l'**inhibition** ou la **saturation** (limitation) des poids, viennent alors **réguler** la dynamique du réseau. La section (2.4.4) se focalise sur ce **rôle** de l'inhibition et de la saturation, en commençant (2.4.4.1) par une **comparaison** avec les modèles classiques d'**inhibition latérale** chez Amari ou Grossberg.

#### 2.4.4.1. Comparaison avec les mécanismes d'inhibition latérale chez Amari ou Grossberg

Dans la littérature sur les réseaux neuronaux, les travaux d'**Amari** et de **Grossberg** constituent des références majeures pour comprendre comment un réseau peut instaurer une dynamique de **compétition** et de **sélection** via des mécanismes d'**inhibition latérale**. Ces modèles, qui sont généralement formulés en temps continu, mettent en œuvre des termes inhibiteurs permettant de restreindre l'activité collective et de favoriser l'émergence de structures distinctes dans le réseau. Il convient dès lors de comparer ces approches classiques avec celles du **Deep Synergy Learning (DSL)**, dans lequel les pondérations  $\omega_{ij}$  évoluent de manière adaptative et peuvent intégrer des termes d'inhibition destinés à limiter la croissance excessive des liaisons.

Dans le modèle d'Amari, par exemple, l'évolution de l'activation  $u_i$  d'une unité est souvent décrite par une équation différentielle de la forme

$$\frac{du_i}{dt} = -u_i + \sum_j w_{ij} f(u_j) - \beta \sum_{j \neq i} g(u_j),$$

où  $w_{ij}$  représente le **couplage excitateur** entre les unités  $i$  et  $j$ ,  $f$  est une fonction d'activation non linéaire, et le terme  $-\beta \sum_{j \neq i} g(u_j)$  incarne l'**inhibition latérale**. Cette inhibition, par son action négative, réduit l'activation des unités voisines lorsqu'une unité donnée devient fortement active, aboutissant ainsi à un phénomène de **compétition** qui peut, par exemple, engendrer des zones d'activité distinctes dans une carte neuronale.

Grossberg, quant à lui, a développé une approche similaire en mettant en évidence la **compétition** entre neurones par le biais d'une répartition limitée de la ressource synaptique. Dans ses modèles, l'évolution de l'activation est soumise à un mécanisme de régulation tel que

$$\frac{du_i}{dt} = -\alpha u_i + \sum_j w_{ij} f(u_j) - \gamma \phi \left( \sum_j w_{ij} f(u_j) \right),$$

où  $\gamma$  et  $\alpha$  sont des constantes positives, et le terme inhibiteur  $-\gamma \phi(\cdot)$  permet de moduler l'activation en fonction de la somme des contributions des autres unités. Ce mécanisme, souvent qualifié de **winner-takes-most**, fait en sorte que seules quelques unités parviennent à conserver une activité significative tandis que les autres sont fortement inhibées.

Dans le **DSL**, l'évolution des pondérations  $\omega_{ij}$  est régie par une mise à jour de type

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta [S(i,j) - \tau \omega_{ij}(t)],$$

où  $\eta$  est le **taux d'apprentissage**,  $\tau$  le **coefficient de décroissance** et  $S(i,j)$  la **synergie effective** entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . Pour intégrer un mécanisme d'**inhibition latérale** similaire à celui des modèles d'Amari et de Grossberg, il est possible d'enrichir la mise à jour en ajoutant un terme inhibiteur explicite, par exemple

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta \left[ S(i,j) - \tau \omega_{ij}(t) - \gamma \sum_{k \neq j} \omega_{ik}(t) \right],$$

où le terme  $-\gamma \sum_{k \neq j} \omega_{ik}(t)$  représente une **pénalisation** de la croissance des autres pondérations sortantes du même nœud  $\mathcal{E}_i$ . Ce mécanisme incite un nœud à concentrer son potentiel de couplage sur un nombre limité de connexions plutôt que de diffuser de manière uniforme ses ressources synaptiques, imitant ainsi la dynamique d'inhibition observée dans les modèles continus d'Amari et Grossberg.

L'avantage principal de cette approche dans le DSL réside dans sa capacité à **sparsifier** le réseau. En effet, l'introduction de l'inhibition latérale force chaque entité à **sélectionner** les liaisons les plus pertinentes, ce qui conduit à la formation de **clusters** distincts et bien définis. Par analogie avec les réseaux de neurones biologiques, ce mécanisme assure que la compétition entre les connexions maintienne une **structure hiérarchisée** du réseau, dans laquelle seuls les liens les plus forts et les plus significatifs persistent, tandis que les connexions moins importantes sont naturellement atténuées.

D'un point de vue mathématique, la similitude avec les modèles d'Amari et Grossberg se reflète dans la forme des équations différentielles. Dans les deux cas, un **terme de décroissance** – qu'il

s'agisse de  $-u_i$  dans l'équation d'Amari ou de  $-\alpha u_i$  dans celle de Grossberg – joue un rôle fondamental en limitant la croissance des variables et en assurant la **stabilité** de la dynamique. De même, dans le DSL, le terme  $-\tau \omega_{ij}(t)$  agit pour contrer l'augmentation induite par la synergie  $S(i, j)$ . La présence du terme inhibiteur supplémentaire  $-\gamma \sum_{k \neq j} \omega_{ik}(t)$  introduit une compétition entre les liaisons sortantes d'un même nœud, ce qui est directement comparable à l'inhibition latérale classique.

En conclusion, la **comparaison** entre les mécanismes d'inhibition latérale d'Amari et de Grossberg et ceux du DSL met en lumière une convergence conceptuelle forte. Alors que les modèles classiques utilisent des termes d'inhibition pour restreindre l'activation globale des unités et favoriser l'émergence de structures distinctes, le DSL exploite des termes analogues dans la mise à jour de ses pondérations pour induire une **compétition** entre les connexions et ainsi favoriser la formation de **clusters** stables et sparsifiés. Cette approche, bien que formulée dans un cadre numérique et évolutif, reprend les principes fondamentaux de la **compétition neuronale** et offre un puissant levier pour la **stabilisation** et la **segmentation** du réseau.

### *Conclusion*

Le parallèle établi avec les mécanismes d'inhibition latérale chez Amari et Grossberg démontre que l'ajout d'un terme inhibiteur dans la mise à jour des pondérations  $\omega_{ij}$  du DSL constitue un régulateur efficace. Ce mécanisme, qui limite la somme des connexions sortantes d'un nœud et favorise la sélection de liens dominants, est en parfaite adéquation avec les principes biologiques d'inhibition latérale observés dans les réseaux neuronaux. Les analogies avec les modèles continus d'Amari et Grossberg offrent ainsi un cadre conceptuel robuste pour expliquer comment la **compétition** entre les liaisons favorise l'**auto-organisation** du réseau en clusters distincts et stables. Les développements ultérieurs exploreront de manière plus approfondie l'impact de ces mécanismes sur la stabilité globale et les analogies avec des modèles biologiques plus complexes.

#### **2.4.4.2. Effet sur la stabilisation d'un nombre limité de clusters ou sur la rupture d'oscillations excessives**

Dans le cadre du **Deep Synergy Learning (DSL)**, l'introduction d'un mécanisme d'**inhibition compétitive** joue un rôle fondamental en régulant l'évolution des pondérations  $\omega_{ij}$ . Cette régulation vise à limiter le nombre de liaisons fortes qu'un même nœud peut entretenir, tout en empêchant l'apparition d'oscillations excessives dans la dynamique du réseau. L'objectif est double : d'une part, forcer la **sélection** de connexions significatives qui se regroupent en clusters distincts et, d'autre part, amortir les rétroactions positives qui pourraient conduire à des oscillations auto-entretenues, voire à des régimes pseudo-chaotiques.

Lorsque l'on considère un DSL sans mécanisme inhibiteur explicite, il est fréquent qu'une entité  $\mathcal{E}_i$  développe de fortes connexions avec un grand nombre d'autres entités  $\mathcal{E}_j$  si la **synergie**  $S(i, j)$  reste suffisamment élevée pour chacune de ces paires. Un tel comportement peut conduire à la formation d'un réseau très dense, dans lequel presque toutes les entités sont fortement couplées, rendant floue l'identification de sous-groupes ou de clusters. Afin de remédier à ce phénomène, l'ajout d'un terme d'inhibition compétitive modifie la règle de mise à jour en intégrant une

pénalisation sur la somme des pondérations sortantes d'un nœud. Par exemple, la mise à jour peut être modifiée comme suit :

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta \left[ S(i,j) - \tau \omega_{ij}(t) - \gamma \sum_{k \neq j} \omega_{ik}(t) \right],$$

où  $\gamma > 0$  représente le **paramètre d'inhibition**. Ce terme additionnel,  $-\gamma \sum_{k \neq j} \omega_{ik}(t)$ , exerce une pression négative sur la croissance simultanée de plusieurs liaisons issues d'un même nœud  $\mathcal{E}_i$ . Ainsi, lorsqu'une pondération  $\omega_{ik}(t)$  croît de manière significative, la somme  $\sum_{k \neq j} \omega_{ik}(t)$  augmente, et par conséquent, la progression de  $\omega_{ij}(t)$  est freinée. Ce mécanisme force chaque entité à « choisir » un nombre limité de connexions fortes, ce qui conduit à la **formation** de clusters distincts et moins nombreux, facilitant ainsi l'**interprétation** de la structure du réseau.

D'autre part, dans les systèmes non linéaires où des rétroactions peuvent induire des oscillations ou des fluctuations exagérées, l'inhibition compétitive agit également comme un **stabilisateur** dynamique. En imposant une contrainte sur la somme totale des liaisons sortantes, le terme inhibiteur contribue à amortir les variations et à réduire la possibilité d'un renforcement en cascade des pondérations qui, autrement, pourrait entraîner des oscillations de grande amplitude. Du point de vue de l'analyse des systèmes dynamiques, cette inhibition se traduit par l'ajout d'un terme négatif dans la **Jacobienne** du système, diminuant ainsi la norme des valeurs propres et favorisant la convergence vers un attracteur stable. Ainsi, la dynamique globale se retrouve mieux équilibrée et la trajectoire des pondérations  $\{\omega_{ij}(t)\}$  se stabilise, limitant les phénomènes de résonance excessive.

En synthèse, l'introduction de l'inhibition dans le DSL présente deux avantages majeurs. Premièrement, elle aboutit à une **sparsification** de la connectivité du réseau en contraignant chaque entité à ne maintenir que quelques liaisons fortes, ce qui rend la formation des clusters plus claire et la segmentation du réseau plus nette. Deuxièmement, elle contribue à **amortir** les oscillations potentielles, en introduisant un mécanisme de retour négatif qui empêche l'exacerbation des rétroactions positives, favorisant ainsi une **stabilité dynamique** accrue. Cette approche, qui s'inspire directement des modèles d'inhibition latérale chez Amari et Grossberg, permet d'obtenir un réseau auto-organisé présentant à la fois une structure hiérarchisée et une robustesse face aux fluctuations internes.

## Conclusion

En incorporant des mécanismes d'inhibition compétitive – par exemple, en ajoutant un terme du type  $-\gamma \sum_{k \neq j} \omega_{ik}(t)$  dans la règle de mise à jour – le DSL parvient à limiter la prolifération des liaisons fortes pour chaque entité, ce qui conduit à la formation d'un nombre restreint et bien défini de clusters. Par ailleurs, cette inhibition joue un rôle crucial dans la rupture des oscillations excessives en amortissant les rétroactions positives, ce qui stabilise la dynamique globale du réseau. Ces effets combinés assurent que la structure auto-organisée demeure à la fois **sparse** et **stable**, permettant ainsi une meilleure interprétation et une exploitation efficace des regroupements observés.

### 2.4.4.3. Discussion : analogies biologiques, applications en robotique sensorielle

L'approche du **Deep Synergy Learning (DSL)**, par l'intermédiaire de ses mécanismes d'inhibition et de saturation, trouve de fortes correspondances avec les observations faites dans la **biologie neuronale**. Dans le cortex sensoriel, par exemple, il est largement démontré que l'**inhibition latérale** joue un rôle déterminant dans la formation de **cartes corticales**. Les neurones fortement activés exercent une inhibition sur leurs voisins, ce qui permet d'isoler des zones d'activité et de créer des **patches** ou des **colonnes d'orientation** dans le cortex visuel. Ce phénomène est essentiel pour la **compétition neuronale** et la spécialisation fonctionnelle, des principes qui ont été théorisés par des auteurs tels qu'**Amari** et **Grossberg**. Ces travaux démontrent que la dynamique d'un réseau de neurones peut être modulée par des mécanismes inhibiteurs qui limitent la propagation de l'excitation et favorisent la sélection d'une **activité dominante** au sein de sous-groupes distincts.

Dans le DSL, la **mise à jour** des pondérations  $\omega_{ij}$  se fait selon une règle qui, lorsqu'enrichie d'un terme inhibiteur tel que

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta \left[ S(i,j) - \tau \omega_{ij}(t) - \gamma \sum_{k \neq j} \omega_{ik}(t) \right],$$

permet de limiter la croissance simultanée de nombreuses connexions issues d'un même nœud  $\mathcal{E}_i$ . Ce terme inhibiteur  $-\gamma \sum_{k \neq j} \omega_{ik}(t)$  reproduit en quelque sorte le principe de la **ressource synaptique limitée** qui est observé dans les systèmes biologiques : lorsqu'une synapse se renforce, les autres connexions du même neurone sont freinées, conduisant à une **sélection** des liens les plus pertinents. Ainsi, cette stratégie favorise l'émergence d'**assemblées neuronales** ou de **clusters** distincts, où seules quelques connexions dominantes persistent, tout en inhibant les connexions superflues.

Du point de vue des **applications en robotique sensorielle**, l'intérêt de ces mécanismes se manifeste de manière très concrète. Dans un système robotique intégrant de multiples capteurs – tels qu'une caméra, un sonar, un LiDAR ou encore des micros – chaque capteur peut être modélisé comme une entité  $\mathcal{E}_i$  au sein d'un SCN. La **synergie**  $S(i,j)$  mesure alors la complémentarité ou la redondance entre deux capteurs. Sans un mécanisme d'inhibition, il est envisageable que le réseau attribue une importance excessive à plusieurs canaux simultanément, ce qui pourrait mener à une **fusion de données** confuse et à une surcharge informationnelle. L'inhibition permet, en revanche, de contraindre le système à privilégier un nombre restreint de connexions robustes – par exemple, en mettant en avant la combinaison d'une vision frontale et d'un microphone ambiant – tout en atténuant l'impact des capteurs moins pertinents à un moment donné. En outre, l'inhibition agit comme un **verrou** contre l'auto-renforcement excessif qui pourrait amplifier indéfiniment le bruit dans certains canaux, garantissant ainsi une **robustesse** accrue du traitement sensoriel.

En somme, cette **discussion** met en lumière l'importance de l'inhibition dans le DSL, tant sur le plan théorique que dans ses **applications pratiques**. L'inhibition, par son effet de **compétition** entre les connexions, permet de réduire la densité des liaisons et de favoriser une **structuration** en clusters, ce qui facilite la segmentation et la hiérarchisation de l'information. Ce mécanisme, qui s'inspire directement des observations biologiques – notamment celles concernant l'inhibition latérale dans le cortex – trouve également une application essentielle dans la robotique sensorielle,

où la gestion efficace des flux de données est cruciale pour une perception fiable et une prise de décision autonome.

### *Conclusion*

Le rôle de l'**inhibition** et de la **saturation** dans un SCN n'est pas uniquement un outil algorithmique, mais il s'inscrit dans une tradition de modèles neuronaux visant à instaurer une compétition synaptique qui permet une **auto-organisation** sélective. Les analogies avec les mécanismes d'inhibition latérale observés dans le cortex et décrits par Amari et Grossberg offrent un cadre conceptuel robuste pour comprendre comment le DSL parvient à stabiliser un nombre limité de clusters tout en évitant les oscillations excessives. Par ailleurs, ces principes se retrouvent également dans des applications concrètes telles que la robotique sensorielle, où ils permettent de gérer de manière efficace et robuste des flux de données complexes et diversifiés. Ainsi, l'intégration de ces mécanismes dans le DSL contribue à la **robustesse** et à la **sélectivité** du réseau, assurant une organisation hiérarchisée et fonctionnelle similaire à celle observée dans les systèmes biologiques.

### *2.4.5. Approches Hybrides et Collaboration Interdisciplinaire*

Tout au long de la section 2.4, nous avons souligné les **connexions** entre le DSL (Deep Synergy Learning) et divers champs : théorie des systèmes dynamiques (2.4.1), modèles de spin (2.4.2) et notion d'énergie potentielle (2.4.3), ou encore mécanismes d'inhibition/saturation (2.4.4). Ces croisements laissent entrevoir de **nombreuses pistes** d'approches dites "hybrides" ou inter-filières, mobilisant la **physique**, la **biologie**, l'**informatique** et l'**ingénierie**. La section 2.4.5 (Approches Hybrides et Collaboration Interdisciplinaire) discute d'abord (2.4.5.1) du **recuit simulé** et d'autres algorithmes inspirés par la **physique statistique** pour régler le SCN, avant de poser (2.4.5.2) des perspectives dans des champs variés comme l'écologie numérique ou les réseaux sociaux, puis de faire (2.4.5.3) une transition vers les chapitres 3, 4, 5 où les mises en forme plus "ingénierie" seront développées en détail.

#### **2.4.5.1. Comment la physique statistique peut fournir des algorithmes inspirés du recuit (simulated annealing) pour régler le SCN**

L'approche du **recuit simulé** en physique statistique repose sur une analogie avec la trempe des métaux, dans laquelle un matériau est chauffé à une température élevée afin de permettre aux atomes de se réarranger librement, puis refroidi lentement pour atteindre une configuration d'énergie minimale. Dans le contexte algorithmique, chaque configuration d'un système est associée à une fonction « énergie »  $\mathcal{H}(\mathbf{x})$  et l'objectif est de minimiser cette énergie en effectuant des transitions aléatoires. La probabilité d'accepter une modification qui augmente l'énergie est donnée par la loi exponentielle

$$P(\Delta\mathcal{H}) \propto \exp\left(-\frac{\Delta\mathcal{H}}{T}\right),$$

où  $T$  représente la **température**. Lorsque  $T$  est élevé, le système explore de manière large l'espace des configurations, et à mesure que  $T$  diminue, les changements qui augmentent  $\mathcal{H}$  deviennent de moins en moins probables, permettant ainsi au système de se stabiliser dans un minimum d'énergie.

Dans le **Deep Synergy Learning (DSL)**, il est naturel de chercher à transposer ce principe au réglage des pondérations  $\omega_{ij}$  du **Synergistic Connection Network (SCN)**. Pour ce faire, on définit une **pseudo-énergie**  $\mathcal{J}(\Omega)$  fonctionnelle des pondérations, dont une expression heuristique pourrait être donnée par

$$\mathcal{J}(\Omega) = - \sum_{i,j} \omega_{ij} S(i,j) + \frac{\tau}{2} \sum_{i,j} \omega_{ij}^2 + \dots,$$

où  $S(i,j)$  désigne la **synergie effective** entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ , et  $\tau$  est un **coefficient de décroissance** qui sert de régularisateur pour éviter une croissance non bornée des poids. Dans un cadre idéal, si l'on considère que  $S(i,j)$  est constant ou ne dépend pas fortement de  $\omega_{ij}$ , la dynamique des pondérations peut être vue comme une **descente de gradient** de  $\mathcal{J}$  :

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \eta \frac{\partial \mathcal{J}}{\partial \omega_{ij}} \Big|_{\omega_{ij}(t)},$$

ce qui, en posant

$$\frac{\partial \mathcal{J}}{\partial \omega_{ij}} = -S(i,j) + \tau \omega_{ij},$$

se traduit par

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta [S(i,j) - \tau \omega_{ij}(t)].$$

Ce processus, qui dans ce cas particulier correspond à une descente d'énergie locale, permet d'expliquer la formation des clusters en considérant que le système tend à minimiser la fonction  $\mathcal{J}$ . L'injection de **bruit** dans ce mécanisme – par exemple en modifiant la mise à jour comme suit

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \eta \frac{\partial \mathcal{J}}{\partial \omega_{ij}} \Big|_{\omega_{ij}(t)} + \sigma \xi_{ij}(t),$$

où  $\xi_{ij}(t)$  est une variable aléatoire et  $\sigma$  en contrôle l'amplitude – permet alors d'implémenter une procédure de **recuit simulé**. Dans cette approche, une température effective  $T(t)$  (ou une amplitude de bruit  $\sigma(t)$ ) est introduite et décroît progressivement. Au départ, le système, exposé à un niveau de bruit élevé, peut explorer divers minima locaux en acceptant occasionnellement des mouvements qui augmentent la pseudo-énergie  $\mathcal{J}$ . Au fur et à mesure que  $T(t)$  décroît, ces mouvements deviennent moins probables et le système se stabilise dans une configuration optimisée.

Il convient de noter que, dans des scénarios plus complexes, où la synergie  $S(i,j)$  dépend de manière non linéaire des pondérations  $\omega_{ij}$  ou de paramètres externes, le paysage d'énergie  $\mathcal{J}(\Omega)$  ne sera pas fixe mais évoluera dans le temps. Dans ce cas, l'application rigoureuse d'un algorithme de recuit simulé devient plus délicate, et la descente d'énergie n'est qu'une **approximation** de la

dynamique réelle du système. Cependant, même dans ces conditions, l'ajout contrôlé de bruit peut aider le SCN à **échapper** à des minima superficiels et à explorer un espace de solutions plus large.

### Conclusion

En définitive, les algorithmes inspirés de la **physique statistique**, tels que le recuit simulé, fournissent un cadre puissant pour réguler la dynamique d'un SCN. En définissant une pseudo-énergie

$$\mathcal{J}(\Omega) = - \sum_{i,j} \omega_{ij} S(i,j) + \frac{\tau}{2} \sum_{i,j} \omega_{ij}^2 + \dots,$$

et en appliquant une mise à jour stochastique de type

$$\omega_{ij}(t+1) = \omega_{ij}(t) - \eta \frac{\partial \mathcal{J}}{\partial \omega_{ij}} \big|_{\omega_{ij}(t)} + \sigma(t) \xi_{ij}(t),$$

on parvient à introduire une **température** contrôlée qui, en diminuant progressivement, permet au réseau d'explorer l'espace des configurations tout en évitant le piégeage dans un minimum local non optimal. Cette approche, bien qu'elle repose sur l'hypothèse d'une synergie relativement stationnaire, représente un levier essentiel pour optimiser la structure des pondérations et, par extension, la formation de clusters dans le DSL.

#### 2.4.5.2. Perspectives d'usage en écologie numérique, réseaux sociaux évolutifs, etc.

La démarche du **Deep Synergy Learning (DSL)**, qui s'appuie sur des concepts issus de la physique statistique tels que le recuit simulé, l'approche énergétique et les mécanismes d'inhibition, peut être étendue bien au-delà du domaine des réseaux neuronaux classiques ou de l'optimisation algorithmique. En effet, ces techniques trouvent également un écho dans des domaines où l'on traite de **réseaux dynamiques** et de systèmes multi-entités, notamment dans l'**écologie numérique** et les **réseaux sociaux évolutifs**. Cette section présente une analyse détaillée de la manière dont la vision du DSL peut contribuer à l'**organisation**, à l'**adaptation** et à la **convergence** de systèmes complexes, soumis à d'importantes évolutions structurelles et temporelles.

Dans le cadre de l'**écologie numérique**, on considère un ensemble d'entités logicielles – par exemple, des microservices, des agents autonomes ou des protocoles distribués – qui interagissent de manière continue pour former un écosystème virtuel. Chaque entité, notée  $\mathcal{E}_i$ , est associée à des caractéristiques spécifiques et à une dynamique de mise à jour de ses liens, représentés par des pondérations  $\omega_{ij}$ . La **synergie**  $S(i,j)$  entre deux entités évalue la valeur ajoutée de leur interaction, qu'il s'agisse d'un gain en performance, en complémentarité fonctionnelle ou en compatibilité technique. Dans ce contexte, le DSL permet de modéliser les interactions comme une forme d'**auto-organisation** où, par des mises à jour locales inspirées de mécanismes de recuit simulé, seules les connexions pertinentes se renforcent tandis que les autres s'affaiblissent progressivement. Par exemple, si l'on définit un opérateur de mise à jour du type

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta [S(i,j) - \tau \omega_{ij}(t)],$$

la dynamique résulte en la formation de sous-groupes d'entités fortement couplées, ce qui se traduit par l'émergence de **communautés** de microservices. La présence de mécanismes d'inhibition, qui



limitent la somme des pondérations sortantes d'un même nœud, assure une **sparsification** du réseau et empêche la sur-connexion généralisée. Ainsi, dans une plateforme distribuée, la structure auto-organisée obtenue facilite la coordination entre services, permettant une répartition efficace des tâches et une meilleure gestion des ressources, sans nécessiter de contrôleur centralisé.

D'un autre côté, dans le domaine des **réseaux sociaux évolutifs**, les utilisateurs – modélisés comme des entités  $\mathcal{E}_i$  – interagissent par le biais de liens dont l'intensité, représentée par  $\omega_{ij}$ , reflète la fréquence ou la qualité des interactions (comme la communication, le partage de contenu ou l'affinité d'opinion). La **synergie**  $S(i, j)$  peut être dérivée d'indices tels que la similarité des centres d'intérêt, le nombre de messages échangés ou encore la co-occurrence d'activités. En intégrant des mécanismes d'inhibition dans la mise à jour des liens, le DSL permet de modéliser la **sélectivité relationnelle** qui caractérise les réseaux sociaux : un utilisateur ne peut entretenir simultanément des connexions fortes avec un très grand nombre d'autres, ce qui conduit à la formation de **communautés** stables. La règle de mise à jour

$$\omega_{ij}(t+1) = \omega_{ij}(t) + \eta \left[ S(i, j) - \tau \omega_{ij}(t) - \gamma \sum_{k \neq j} \omega_{ik}(t) \right]$$

illustre comment le terme inhibiteur  $-\gamma \sum_{k \neq j} \omega_{ik}(t)$  force l'utilisateur à concentrer ses interactions sur un sous-ensemble d'autres utilisateurs, reproduisant ainsi l'effet de **compétition synaptique** observé dans le cerveau. Ce mécanisme aide à prévenir la dispersion des ressources relationnelles et contribue à la formation de clusters sociaux, où les interactions internes sont beaucoup plus intenses que celles avec le reste du réseau.

Les **perspectives d'usage** du DSL dans ces domaines sont multiples. Dans l'**écologie numérique**, par exemple, l'auto-organisation des microservices peut permettre de construire des écosystèmes résilients où l'interopérabilité est constamment optimisée par des mises à jour locales. Chaque service ajuste ses connexions en fonction de la synergie mesurée avec d'autres services, et l'ensemble du système évolue vers une configuration optimale qui minimise les coûts de communication et maximise la performance globale. De même, dans les **réseaux sociaux évolutifs**, la dynamique du DSL peut aider à prédire et à comprendre l'évolution des communautés. La capacité du système à s'adapter à l'arrivée ou au départ d'utilisateurs, combinée aux mécanismes de recuit simulé et d'inhibition, permet d'obtenir une modélisation robuste de la formation, de la fusion ou de la fragmentation des groupes sociaux, tout en tenant compte des fluctuations inhérentes aux interactions humaines.

Enfin, l'approche DSL, grâce à sa **flexibilité** dans la définition de la fonction de synergie  $S(i, j)$  et de la mise à jour des pondérations, offre un cadre général capable de s'adapter à de nombreux autres domaines. Que ce soit pour modéliser des réseaux d'innovation, des collaborations scientifiques ou des systèmes logistiques distribués, l'analogie avec les techniques d'**optimisation stochastique** et de recuit simulé reste pertinente. Les méthodes DSL permettent alors de gérer des **interactions** complexes et de produire des configurations stables dans un environnement évolutif, favorisant ainsi l'émergence de structures cohérentes à partir d'un ensemble initialement désordonné.

## Conclusion

Les approches inspirées de la physique statistique, notamment le recuit simulé, confèrent au DSL une capacité interdisciplinaire remarquable. En appliquant ces techniques à des domaines tels que l'écologie numérique et les réseaux sociaux évolutifs, le DSL permet de modéliser l'auto-organisation de systèmes complexes où les entités interagissent de manière dynamique et adaptative. Dans ces contextes, la **flexibilité** du DSL se traduit par une mise à jour auto-adaptative des liaisons, capable de gérer les fluctuations inhérentes aux environnements évolutifs, tout en favorisant la formation de **clusters** robustes et pertinents. Cette vision ouvre la voie à de nombreuses applications concrètes, allant de la gestion distribuée des microservices dans le cloud aux mécanismes de segmentation et d'évolution des communautés dans les réseaux sociaux, en passant par des systèmes collaboratifs dans divers domaines d'innovation.

### 2.4.5.3. Transition vers les Chapitres 3, 4, 5, où l'on verra des mises en forme plus "ingénierie" du SCN

Dans les sections précédentes, nous avons établi un cadre théorique robuste pour le **Deep Synergy Learning (DSL)** en nous appuyant sur des analogies avec la **physique statistique**, les **modèles de spin** (tels que Ising, Potts ou Hopfield) ainsi que sur des concepts issus de la **théorie des systèmes dynamiques** et des **mécanismes d'inhibition** observés en neurosciences. La formulation du DSL repose sur une dynamique de pondérations  $\{\omega_{ij}\}$  évoluant suivant une règle de mise à jour qui, dans sa version continue, s'exprime par l'équation différentielle

$$\frac{d\omega_{ij}}{dt} = \eta [S(i, j; \{\omega_{p,q}\}, \dots) - \tau \omega_{ij}],$$

où  $\eta > 0$  désigne le **taux d'apprentissage** et  $\tau > 0$  le **coefficient de décroissance**. Les développements théoriques des sections 2.4.1 à 2.4.5 mettent en exergue des idées telles que la **descente d'énergie** dans un paysage complexe, l'**inhibition compétitive** rappelant les mécanismes d'Amari et de Grossberg, ainsi que l'utilisation de techniques de **recuit simulé** pour éviter le piégeage dans des minima locaux. Ces concepts, tout en étant profondément ancrés dans des modèles théoriques multidisciplinaires, ouvrent la voie à une transposition vers une **implémentation opérationnelle**.

Il convient désormais de passer de cette **maquette conceptuelle** à une **ingénierie concrète**. C'est exactement l'objectif des Chapitres 3, 4 et 5, qui se proposent d'apporter une déclinaison pratique et opérationnelle de l'approche DSL. Plus précisément, le **Chapitre 3** détaillera la structuration du SCN en exposant les **structures de données** nécessaires pour gérer les matrices de pondérations  $\{\omega_{ij}\}$  et les **algorithmes** de mise à jour, qu'ils soient de type **batch**, **en ligne** ou **stochastique**. On y développera notamment des stratégies pour la gestion de seuils (notion de  $\omega_{\min}$ ) et l'implémentation de mécanismes d'inhibition, en s'appuyant sur des formulations mathématiques précises.

Le **Chapitre 4** se concentrera quant à lui sur les aspects **logiciels** et **parallélisme computationnel**. Il exposera les environnements de développement adaptés – qu'il s'agisse de langages comme Python pour la flexibilité ou de C++/CUDA pour la performance – et décrira comment répartir le calcul sur plusieurs cœurs ou sur des architectures GPU. La modularisation des fonctions, par exemple la séparation entre le module de calcul de la synergie  $S(i, j)$  et le module de mise à jour

des pondérations, sera abordée en détail, permettant ainsi d'envisager une implémentation robuste et scalable du SCN.

Enfin, le **Chapitre 5** portera sur la **validation** et l'**évaluation** des performances du SCN dans des scénarios réels. Il s'agira de définir des **critères de convergence** (par exemple, lorsque  $\|\omega_{ij}(t+1) - \omega_{ij}(t)\|$  devient négligeable), d'évaluer la **cohésion** des clusters formés (via des indicateurs comme la modularité ou le ratio interne/externe) et de mener des expériences sur des jeux de données multimodaux ou issus de flux sensoriels. Ces tests permettront de vérifier la robustesse, la **résilience** face au bruit et la **scalabilité** des approches développées.

En résumé, le passage de la théorie à l'ingénierie concrète se matérialise par une série de développements progressifs. Le **Chapitre 2** a ainsi posé les **fondements** conceptuels et mathématiques du DSL, en s'appuyant sur des analogies interdisciplinaires. La transition vers les **Chapitre 3**, **Chapitre 4** et **Chapitre 5** marquera l'étape où ces idées abstraites se déclinent en algorithmes opérationnels, en architectures logicielles et en méthodologies d'évaluation pratiques, ouvrant la voie à des applications réelles dans des domaines aussi variés que la **robotique**, le **cloud computing**, l'**écologie numérique** ou les **réseaux sociaux évolutifs**. Ce cheminement vise à démontrer qu'un cadre théorique solide peut être traduit en solutions ingénierie efficaces, capables de gérer la complexité et la dynamique d'un système multi-entités en constante évolution.

## 2.5. Perspectives Historiques et Liens avec l'IA Moderne

Au terme de ce **Chapitre 2**, nous avons exploré les **fondements théoriques** du DSL (Deep Synergy Learning) : liens avec la théorie des systèmes dynamiques, rapprochements avec la physique statistique, mécanismes d'inhibition et de saturation, etc. Il est intéressant de **resituer** tout cela dans un **contexte historique** et de voir comment le DSL peut **converger** ou s'articuler avec les **approches** actuelles de l'IA (apprentissage profond, IA neuro-symbolique, robotique adaptative, etc.). La section (2.5) propose un **regard** à la fois **rétrospectif** (comment ces idées se sont forgées) et **prospectif** (comment elles se connectent à l'IA moderne), préparant ainsi la transition vers les chapitres plus “implémentation” et “application”.

### 2.5.1. Évolutions Récentes et Convergence avec le Deep Learning

Depuis la vague d'**apprentissage profond** (Deep Learning) qui domine l'IA contemporaine (vision, langage, RL, etc.), l'idée de laisser des **couches** apprendre automatiquement des **features** est devenu un paradigme central. Dans cette optique, on peut se demander si un **SCN** (Synergistic Connection Network) ne pourrait pas venir **compléter** ou **fusionner** avec des réseaux neuronaux profonds, de façon à gérer l'**auto-organisation** à un niveau plus abstrait.

#### 2.5.1.1. Essor de l'apprentissage profond : extraire des features et laisser un SCN auto-organiser ces caractéristiques à un niveau plus abstrait ?

Dans le contexte actuel de l'**apprentissage profond**, il est devenu courant d'utiliser des réseaux neuronaux tels que les **CNN**, **RNN** ou **Transformers** afin d'extraire automatiquement des **représentations** riches à partir de données brutes telles que des images, des signaux sonores ou des textes. Ces réseaux, souvent désignés comme des **backbones**, produisent des embeddings de haute dimension, par exemple un vecteur  $\mathbf{x}_i \in \mathbb{R}^{512}$  pour chaque donnée d'entrée. L'idée qui se dégage est de ne pas se contenter d'envoyer ces features directement dans un classifieur supervisé, mais d'exploiter l'**auto-organisation** d'un **Synergistic Connection Network (SCN)** pour traiter ces caractéristiques à un niveau d'abstraction supérieur.

Dans ce cadre, chaque entité  $\mathcal{E}_i$  issue du backbone représente un ensemble de features qui sont ensuite transmises au SCN. La **fonction de synergie**  $S(i, j)$  quantifie la relation ou la **complémentarité** entre les représentations de deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . À partir de là, les pondérations  $\omega_{ij}$  évoluent selon une dynamique d'**auto-organisation** décrite par une règle de mise à jour (voir par exemple l'équation en temps discret) :

$$\omega_{ij}(t + 1) = \omega_{ij}(t) + \eta [S(i, j) - \tau \omega_{ij}(t)],$$

où  $\eta > 0$  représente le **taux d'apprentissage** et  $\tau > 0$  le **coefficient de décroissance**. Ce processus permet au SCN d'explorer l'espace latent des features et de faire émerger des **clusters synergiques** qui ne seraient pas immédiatement décelables via une simple classification. En d'autres termes, plutôt que de procéder à une classification rigide, le SCN se charge de **combiner** et d'**organiser** ces embeddings en regroupements cohérents qui révèlent des structures plus abstraites et potentiellement pertinentes pour l'application considérée.

Un schéma typique de pipeline illustrant ce processus s’articule ainsi :

Backbone (ResNet, BERT, etc.) → Feature Embedding  
→ SCN (pondérations  $\omega_{ij}$  évolutives).

Dans ce pipeline, le module de **feature extraction** produit des représentations vectorielles de qualité, qui sont ensuite traitées par le SCN. Ce dernier module peut fonctionner de manière **complètement non supervisée**, en laissant la dynamique locale – régulée par la synergie  $S(i, j)$  et éventuellement enrichie par des mécanismes d’inhibition et de recuit simulé – conduire à la formation de clusters. Alternativement, il peut coexister avec des modules supervisés qui valident la structure émergente des regroupements, offrant ainsi une approche hybride permettant de bénéficier à la fois de la puissance de l’**apprentissage profond** et de la capacité d’**auto-organisation** du SCN.

L’avantage principal de cette approche réside dans sa **flexibilité**. En effet, la capacité du SCN à se réorganiser dynamiquement permet d’extraire des **patterns** inattendus ou des regroupements latents, qui peuvent correspondre à des catégories ou des classes qui ne sont pas explicites dans les données d’origine. Par exemple, dans le traitement d’un flot d’images par un ResNet, le SCN pourrait découvrir des **proto-classes** ou des regroupements fondés sur des similarités spatiales et contextuelles que l’on ne retrouverait pas dans un classifieur standard. Cette capacité à explorer et à structurer l’espace latent des features offre une plus-value significative en termes de **représentation** et de **compréhension** des données, ouvrant la voie à des applications avancées en vision par ordinateur, en traitement du langage naturel et en robotique.

Historiquement, certains systèmes expérimentaux ont déjà exploité des **modules compétitifs** et des **couches hebbiennes** pour favoriser la spécialisation des neurones dans des architectures précurseurs du deep learning moderne. Le DSL reprend et étend ces idées en introduisant une dimension d’auto-organisation qui opère non seulement sur les activations, mais directement sur les **liaisons** entre entités. Ainsi, au lieu de modifier uniquement les sorties des neurones, le DSL permet aux poids  $\omega_{ij}$  de s’ajuster pour favoriser des interactions de plus haut niveau, rendant possible l’émergence d’une **structure hiérarchisée** dans l’espace des features.

## *Conclusion*

En conclusion, l’essor de l’**apprentissage profond** offre la possibilité de combiner les approches d’extraction de **features** à haute dimension avec un SCN capable d’**auto-organiser** ces caractéristiques en regroupements synergiques. Cette approche hybride permet de transcender la simple classification en exploitant une **dynamique** de pondérations  $\omega_{ij}$  qui, grâce à des mécanismes inspirés du recuit simulé et des modèles de compétitions synaptiques, peut révéler des structures abstraites et des relations complexes au sein de l’espace latent. Les **chapitres suivants** (notamment les Chapitres 8 et 14) développeront des scénarios d’application concrets où cette fusion DSL + Deep Learning s’avère particulièrement efficace pour la classification multi-domaine, la détection de patterns temporels complexes et l’organisation de flux multimodaux, démontrant ainsi la pertinence de cette approche dans des domaines variés et en constante évolution.

### 2.5.1.2. Exemples précurseurs : modules compétitifs, couches “Hebbiennes” dans certaines architectures de recherche

Dans l'histoire de l'intelligence artificielle, bien avant l'avènement généralisé du **Deep Learning** tel qu'on le connaît aujourd'hui et avant la formulation explicite du **Deep Synergy Learning (DSL)**, de nombreux chercheurs ont exploré des mécanismes d'apprentissage non supervisé fondés sur la compétition et l'auto-organisation. Ces travaux précurseurs, qui incluent les **modules compétitifs** et les **couches Hebbiennes**, ont permis de poser les jalons d'une approche qui, dans sa version moderne, se retrouve incarnée dans le DSL. On peut ainsi considérer ces premières architectures comme des prototypes conceptuels ayant anticipé la logique d'un **Synergistic Connection Network (SCN)**.

Dans les années 1980 et 1990, le paradigme du **Competitive Learning** fut largement étudié. L'idée fondamentale était d'amener un ensemble de neurones, ou de prototypes, à « compétition » pour représenter des caractéristiques d'une donnée d'entrée. Les algorithmes tels que le **Learning Vector Quantization (LVQ)** reposaient sur le principe que, pour une donnée d'entrée  $\mathbf{x}$ , le prototype le plus proche (selon une métrique souvent euclidienne) recevait un renforcement de son vecteur de poids. Mathématiquement, la règle de mise à jour pour le prototype gagnant, disons  $\mathbf{w}_i$ , pouvait s'exprimer de manière simplifiée par

$$\Delta \mathbf{w}_i = \eta (\mathbf{x} - \mathbf{w}_i),$$

ce qui tend à aligner  $\mathbf{w}_i$  sur les vecteurs d'entrée similaires. Ce mécanisme, par son **aspect compétitif**, impose une différenciation entre les prototypes et favorise la formation de **clusters** d'entrées. La compétition ainsi instaurée conduit naturellement à une **auto-organisation** dans l'espace de représentation, préfigurant la dynamique d'un SCN où les pondérations entre entités s'ajustent en fonction d'une **synergie** mesurée.

Parallèlement, les **Self-Organizing Maps (SOM)** de Kohonen, introduites dans les années 1980, ont apporté une dimension topologique à cette idée. Dans un SOM, chaque neurone de la carte possède un vecteur de poids et la mise à jour se fait de manière à non seulement adapter le neurone gagnant à l'entrée, mais aussi à ajuster ses voisins dans le voisinage défini par un **noyau d'inhibition**. La règle de mise à jour pour un neurone voisin peut être formulée par

$$\Delta \mathbf{w}_j = \eta h(i, j) (\mathbf{x} - \mathbf{w}_j),$$

où  $h(i, j)$  représente une fonction décroissante en fonction de la distance topologique entre le neurone gagnant  $i$  et le neurone  $j$ . Ce mécanisme permet de préserver la **cohérence spatiale** dans la carte, et il a servi de modèle pour comprendre comment un réseau peut se partitionner en régions distinctes d'activité, concept qui trouve une résonance dans la formation de clusters dans le DSL.

Le **postulat Hebbien** de Donald Hebb, formulé en 1949, constitue également une pierre angulaire dans cette évolution conceptuelle. Le célèbre adage « **Neurons that fire together, wire together** » a inspiré les premières règles d'apprentissage synaptique. La règle classique de Hebb, qui peut être écrite comme

$$\Delta w_{ij} \propto x_i x_j,$$

implique que si deux neurones sont activés simultanément, le lien synaptique entre eux se renforce. Bien que cette règle ne comporte pas de mécanisme intrinsèque de régulation (ce qui peut conduire

à une croissance indéfinie des poids), des modifications ultérieures, telles que l'ajout d'un terme de normalisation ou d'inhibition, ont permis de stabiliser ces dynamiques. Ces ajustements ont donné naissance à des **couches Hebbiennes** expérimentales dans certaines architectures de recherche, où l'idée de renforcer les connexions en fonction de la co-activation était exploitée pour générer des représentations auto-organisées des données.

Des approches connexes, telles que l'**Adaptive Resonance Theory (ART)** développée par Stephen Grossberg et Gail Carpenter, combinent des mécanismes de compétition, de vigilance et de reset pour permettre à un réseau de classer de manière stable tout en intégrant de nouvelles informations sans oublier les anciennes. Les modèles ART illustrent comment la dynamique compétitive et les processus de **stabilisation** peuvent coexister dans des systèmes non supervisés, ce qui préfigure la logique d'un SCN dans le cadre du DSL.

En résumé, les **modules compétitifs** et les **couches Hebbiennes** ont constitué des **précurseurs** essentiels dans l'évolution de l'apprentissage non supervisé. Ces systèmes ont démontré que la **compétition** entre neurones – conduisant à la formation de clusters d'activité – ainsi que l'**apprentissage Hebbien** pouvaient engendrer une auto-organisation efficace des connexions. Le DSL s'inscrit dans cette lignée en généralisation, en permettant aux pondérations  $\omega_{ij}$  d'évoluer selon une **fonction de synergie**  $S(i, j)$  qui peut intégrer diverses mesures (distance, co-information, etc.) et en incluant des mécanismes de régulation tels que la **décroissance** (via un terme  $\tau \omega_{ij}$ ) ou l'**inhibition compétitive**. Ainsi, l'approche DSL synthétise et étend les idées anciennes en les adaptant aux défis modernes de la **multimodalité** et de l'**apprentissage continu**, grâce notamment aux avancées récentes en calcul (GPU, big data) qui permettent de gérer des réseaux de grande dimension.

## *Conclusion*

Les travaux antérieurs sur les **modules compétitifs** et les **couches Hebbiennes** offrent une base conceptuelle et expérimentale riche qui préfigure la logique du DSL. Ces architectures précurseurs, en instaurant une dynamique de renforcement des liens basée sur la co-activation et la compétition, ont démontré la faisabilité d'une auto-organisation des connexions dans un réseau. Le DSL généralise cette approche en permettant aux pondérations  $\omega_{ij}$  de s'ajuster selon une **fonction de synergie** plus complexe, intégrant des aspects non linéaires et des mécanismes de régulation supplémentaires. En définitive, ces exemples historiques constituent un tremplin vers des architectures modernes où l'auto-organisation et la dynamique compétitive sont exploitées de manière plus fine, ouvrant la voie à des systèmes adaptatifs et multimodaux d'une grande robustesse et d'une forte capacité d'apprentissage.

### *2.5.2. Extensions Symboliques et Neuro-Symboliques*

Dans la lignée des **perspectives historiques** (2.5.1), on constate que le **DSL** (Deep Synergy Learning) ne se limite pas à des entités purement numériques ou strictement sub-symboliques : le cadre théorique du DSL étant très **général**, il peut **intégrer** à la fois des représentations **logiques** (symboliques) et des composantes **sub-symboliques** (vecteurs, embeddings, etc.). La section (2.5.2) aborde ces **extensions symboliques et neuro-symboliques**, montrant que le DSL offre un terrain d'entente entre l'IA symbolique (règles, concepts, raisonnements) et l'IA sub-symbolique

(réseaux neuronaux, traitement statistique). Nous commençons (2.5.2.1) par rappeler que le DSL peut accueillir des entités logiques ou sub-symboliques en toute flexibilité.

### 2.5.2.1. Rappel : DSL peut accueillir des entités logiques ou des blocs sub-symboliques

Dans le cadre du **Deep Synergy Learning (DSL)**, la conception des « entités » n'est pas nécessairement limitée à des vecteurs continus appartenant à un espace  $\mathbb{R}^d$ . En effet, l'architecture d'un **Synergistic Connection Network (SCN)** se caractérise par sa grande **flexibilité** quant à la nature des entités qui le composent. Dès lors, il est possible d'intégrer dans un même réseau à la fois des **entités sub-symboliques** et des **entités logiques** ou des blocs **symboliques**, ce qui permet de traiter simultanément des représentations continues et des représentations discrètes.

Dans la perspective **sub-symbolique**, les entités  $\mathcal{E}_i$  sont généralement représentées par des **embeddings**, c'est-à-dire des vecteurs caractéristiques issus de réseaux profonds (tels que CNN, RNN, ou Transformers). Ainsi, une entité est formellement décrite par un vecteur

$$\mathbf{x}_i \in \mathbb{R}^d,$$

et la **synergie** entre deux entités, notée  $S(i, j)$ , est calculée à l'aide de mesures classiques telles que la **distance euclidienne** ou la **similarité cosinus**. Par exemple, une fonction de synergie peut être définie comme

$$S(i, j) = \exp(-\alpha \|\mathbf{x}_i - \mathbf{x}_j\|),$$

où  $\alpha > 0$  est un paramètre de sensibilité. Cette formulation permet d'extraire des caractéristiques robustes issues de données brutes (images, sons, textes, etc.) et d'auto-organiser ces représentations en **clusters** par la suite.

D'un autre côté, le DSL permet également d'accueillir des **entités logiques** ou des **blocs symboliques**. Dans ce cas, une entité  $\mathcal{E}_i$  peut être définie non pas par un simple vecteur, mais par un objet conceptuel ou un ensemble de règles. Par exemple, on peut envisager que  $\mathcal{E}_i$  représente un **concept** ou une **proposition** formelle, telle que

$$\mathcal{E}_i = \text{"Si } A \text{ alors } B\text{"}.$$

La fonction de synergie  $S(i, j)$  dans ce contexte sera alors adaptée pour quantifier la **compatibilité** ou la **cohérence** entre deux ensembles de règles ou entre des concepts issus d'ontologies différentes. Une approche possible consiste à définir une distance sémantique ou une mesure de **similarité logique** basée sur la structure de l'ontologie. Par exemple, on pourrait utiliser une mesure de similarité définie par

$$S(i, j) = \sigma(\mathcal{E}_i, \mathcal{E}_j) \in [0, 1],$$

où  $\sigma$  est construite à partir d'une métrique de correspondance sémantique.

L'**intégration** de ces deux types d'entités dans un même SCN permet de bâtir un modèle **neuro-symbolique** puissant, capable de combiner les avantages du traitement continu (apprentissage à partir de données brutes, robustesse aux variations) et ceux du raisonnement symbolique (interprétabilité, manipulation explicite de concepts et règles). L'une des **forces** du



DSL réside dans sa capacité à utiliser une **fonction de synergie**  $S(i, j)$  qui peut être spécifiquement adaptée en fonction de la nature des entités concernées. Ainsi, pour des entités sub-symboliques, on utilisera des mesures issues de la **géométrie de l'espace vectoriel** (distance euclidienne, similarité cosinus, etc.), tandis que pour des entités logiques, des techniques de **matching sémantique** ou des calculs de **coïncidence symbolique** seront privilégiées.

Cette approche hybride permet d'aboutir à une **auto-organisation** qui se fonde sur une **complémentarité** entre des représentations denses et continues et des représentations structurées et discrètes. Les pondérations  $\omega_{ij}$  qui relient les entités dans le SCN s'ajustent ainsi en fonction d'une **synergie** multi-dimensionnelle, où chaque lien reflète à la fois une similitude perceptuelle et une compatibilité conceptuelle. L'ensemble du réseau, noté

$$\Omega = \{\omega_{ij}\},$$

se structure de manière à mettre en avant les connexions les plus **pertinentes** et à minimiser celles qui sont moins significatives, quelle que soit la nature des entités.

### *Conclusion*

En résumé, le **DSL** se distingue par sa capacité à intégrer un **mélange** d'entités, aussi bien **sub-symboliques** que **logiques**. Cette flexibilité est précieuse car elle permet d'aborder des environnements **multimodaux** où coexistent des données continues (images, sons, textes) et des informations symboliques (catégories, règles, concepts). La **fonction de synergie**  $S(i, j)$  est alors conçue pour s'adapter à ces différents types de représentations, assurant ainsi que le SCN puisse auto-organiser ses connexions en tenant compte à la fois des **aspects perceptifs** et **conceptuels**. Cette approche ouvre la voie à des applications neuro-symboliques avancées, où la **plasticité** du réseau exploite pleinement la richesse des données d'entrée pour produire des **clusters** d'entités qui reflètent des relations complexes et complémentaires.

#### **2.5.2.2. Synergies entre IA symbolique, sous-ensembles de règles, et la plasticité d'un SCN**

Dans cette section, nous explorons comment l'**IA symbolique** – qui s'appuie sur la manipulation de règles, de concepts et d'ontologies – peut interagir de manière fructueuse avec la **plasticité** inhérente à un **Synergistic Connection Network (SCN)**, tel que présenté dans le cadre du **DSL** (Deep Synergy Learning). Cette synergie repose sur la capacité du SCN à adapter dynamiquement les pondérations  $\omega_{ij}$  en fonction d'une fonction de synergie  $S(i, j)$  adaptée aux différentes natures d'entités. Nous nous référons ici aux idées développées dans les sections précédentes, notamment aux notions d'auto-organisation évoquées en (2.5.2.1) et aux fondements théoriques du DSL présentés dans les sections 2.3 et 2.4.

##### *A. IA symbolique et “blocs de règles” : rappel et enjeux*

Historiquement, l'**IA symbolique** s'est concentrée sur la manipulation de symboles, de règles formelles et de représentations logiques. Dans ce paradigme, une entité symbolique peut être un **bloc de règles** ou un **concept** défini par une structure logique, par exemple une proposition du type

if  $A$  then  $B$ .

Ces entités disposent d'une **force interne** basée sur la validité ou la cohérence de leurs règles et, lorsqu'elles interagissent avec d'autres blocs similaires, une mesure de **compatibilité** peut être définie. Par exemple, si deux ensembles de règles  $\mathcal{R}_1$  et  $\mathcal{R}_2$  se complètent, leur synergie peut être quantifiée par une fonction de similarité, que nous noterons  $S_{\text{sym}}(i, j)$ . Ce mécanisme est similaire aux systèmes d'inférence dans l'IA symbolique classique, où un moteur de règles applique des opérations logiques pour en déduire de nouvelles connaissances.

### *B. Plasticité d'un SCN : une approche dynamique de l'auto-organisation*

Dans le cadre du DSL, les **entités** ne sont pas exclusivement sub-symboliques (comme les vecteurs  $\mathbf{x}_i \in \mathbb{R}^d$  issus d'un réseau de neurones) ; elles peuvent également être des entités symboliques. La **fonction de synergie**  $S(i, j)$  est alors conçue de manière à prendre en compte des mesures issues tant de la comparaison de vecteurs que de la compatibilité entre ensembles de règles. Par exemple, pour une entité symbolique  $\mathcal{E}_{\text{sym}}$  et une entité sub-symbolique  $\mathcal{E}_{\text{sub}}$ , on peut définir une synergie mixte

$$S(\mathcal{E}_{\text{sym}}, \mathcal{E}_{\text{sub}}) = f\left(\sigma(\mathcal{E}_{\text{sym}}, \mathcal{E}_{\text{sub}})\right),$$

où  $\sigma$  représente une métrique de compatibilité (telle qu'une distance sémantique ou une mesure d'inférence) et  $f$  est une fonction d'activation adaptée. L'idée est que si les données sub-symboliques confirment ou valident le contenu d'un bloc de règles, le lien  $\omega_{ij}$  entre ces deux entités se renforce automatiquement, selon la règle de mise à jour DSL, telle que présentée dans la section (2.2.2) et analysée dans (2.3.1) et (2.4.1).

La **plasticité** du SCN se traduit par l'évolution dynamique des pondérations  $\omega_{ij}$  qui se mettent à jour, par exemple, selon la règle additive

$$\omega_{ij}(t + 1) = \omega_{ij}(t) + \eta [S(i, j) - \tau \omega_{ij}(t)],$$

où  $\eta$  est le taux d'apprentissage et  $\tau$  le terme de décroissance. Lorsque  $S(i, j)$  est une fonction qui intègre à la fois des éléments symboliques et sub-symboliques, la dynamique de  $\omega_{ij}$  permet une **auto-organisation** fine des entités en clusters, reflétant des affinités multiples.

### *C. Interactions entre les couches symbolique et sub-symbolique*

L'une des idées centrales de l'approche neuro-symbolique est de permettre une **cohabitation** entre une couche symbolique, où résident des règles explicites et des concepts, et une couche sub-symbolique, issue des techniques d'apprentissage profond. Dans un SCN, cela se traduit par la coexistence d'entités  $\mathcal{E}_i$  qui peuvent être soit des vecteurs d'activation (issues d'un backbone de deep learning, comme un ResNet ou un Transformer), soit des blocs de règles ou des concepts logiques. Les liens entre ces entités sont modulés par une fonction de synergie  $S(i, j)$  conçue pour comparer des types de données hétérogènes.

Par exemple, dans une application pratique, un bloc de règles décrivant des relations logiques (par exemple, la règle *if animal then vertebrate*) pourra interagir avec des embeddings issus d'images ou de textes. Si un grand nombre d'images correspondant à des animaux se retrouvent alignées avec ce bloc, la synergie mesurée augmentera, ce qui, selon la dynamique du DSL, renforcera la pondération  $\omega_{ij}$  correspondante. Ainsi, le SCN pourra, de manière auto-organisée, former un cluster qui associe les représentations symboliques et sub-symboliques autour d'un thème commun.

Les avantages de cette approche résident notamment dans la **flexibilité** et la **richesse** de l'auto-organisation, car le réseau n'est pas contraint à une seule modalité de représentation. Comme indiqué dans la section (2.5.2.1), l'adaptation aux divers types d'entités permet d'étendre l'application du DSL à des environnements multimodaux, où coexistent des informations perceptuelles et des concepts explicites.

#### *D. Applications et implications de la synergie neuro-symbolique*

Les synergies entre IA symbolique et plasticité d'un SCN ouvrent de nombreux champs d'application :

- **Systèmes experts évolutifs** : Dans ces systèmes, les règles ne sont pas figées mais évoluent en fonction des données. Un SCN peut adapter dynamiquement les pondérations entre différentes règles en fonction de leur performance ou de leur pertinence dans un environnement changeant. Cette approche permet de résoudre le problème de la rigidité des systèmes experts classiques.
- **Raisonnement multimodal** : En intégrant des représentations issues à la fois de sources sub-symboliques (comme des images ou du texte) et symboliques (comme des ontologies ou des règles), le SCN facilite la formation de clusters hybrides capables d'exploiter des informations complémentaires. Par exemple, dans le domaine de la vision par ordinateur, un bloc de règles décrivant des relations spatiales pourrait être combiné avec des embeddings d'images pour améliorer la reconnaissance d'objets.
- **Systèmes neuro-symboliques unifiés** : L'approche permet de construire des architectures où la logique symbolique, avec ses avantages en termes d'interprétabilité et de raisonnement formel, est intégrée dans un cadre d'apprentissage profond, offrant ainsi des solutions plus robustes et adaptatives pour des tâches complexes.

Ces exemples illustrent comment la **plasticité** du SCN, combinée à une **fonction de synergie** capable de traiter des entités hétérogènes, peut conduire à une auto-organisation efficace des connaissances, permettant d'associer de manière dynamique des informations issues de différentes modalités.

#### *Conclusion*

Pour conclure cette sous-section, nous récapitulons les points essentiels en nous référant aux sections antérieures (notamment (2.5.2.1)) :

- **Auto-organisation hybride** : Le DSL permet d'accueillir simultanément des entités sub-symboliques (embeddings, vecteurs de features) et des entités symboliques (blocs de règles, concepts logiques). La fonction de synergie  $S(i,j)$  est adaptée à chaque type d'entité, favorisant ainsi des liens pertinents entre elles.
- **Plasticité et adaptation** : La mise à jour des pondérations  $\omega_{ij}$  via des mécanismes de descente de gradient (ou des versions stochastiques) permet au SCN de s'auto-organiser de manière dynamique. Ainsi, les blocs de règles peuvent se renforcer ou s'affaiblir en fonction de leur compatibilité avec les données sub-symboliques.
- **Perspectives neuro-symboliques** : Cette approche ouvre la voie à des systèmes d'IA capables de combiner le meilleur des deux mondes – la capacité d'apprentissage et d'adaptation des réseaux de neurones et la clarté du raisonnement symbolique – pour aboutir à des architectures plus complètes et robustes.

Nous verrons dans la section (2.5.2.3) comment ces idées seront approfondies dans des **chapitres futurs** (notamment les chapitres 5 et 13), afin de déployer des systèmes neuro-symboliques intégrés qui tirent pleinement parti de la complémentarité entre traitement symbolique et sub-symbolique dans le cadre du DSL.

### 2.5.2.3. Chapitres futurs (5, 13) où l'on détaillera la cohabitation symbolique-subsymbolique

Les développements présentés dans les sections (2.5.2.1) et (2.5.2.2) ont montré que le **DSL** (Deep Synergy Learning) se distingue par sa capacité à accueillir des entités de nature hétérogène, allant des représentations sub-symboliques (vecteurs, embeddings) aux entités symboliques (ensembles de règles, concepts logiques). Cette cohabitation est rendue possible par une fonction de synergie  $S(i,j)$  capable d'évaluer, de manière adaptée, la compatibilité ou la complémentarité entre des entités aux représentations très différentes. Toutefois, afin de passer du cadre théorique à une application concrète et opérationnelle, il est nécessaire de préciser les modalités d'implémentation et d'intégration de ces idées dans un système complet. C'est exactement ce que viseront les **Chapitre 5** et **Chapitre 13** de notre ouvrage.

Dans le **Chapitre 5**, nous aborderons tout d'abord l'**architecture générale** du Synergistic Connection Network (SCN). Ce chapitre détaillera la manière dont les pondérations  $\omega_{i,j}$  sont structurées et stockées (par exemple sous forme de matrices denses ou creuses) ainsi que les différentes stratégies d'**initialisation** et de mise à jour. Nous y décrirons également comment les données d'entrée—qu'elles soient issues d'un réseau de neurones pré-entraîné (représentations sub-symboliques) ou de modules symboliques (ensembles de règles, ontologies)—sont intégrées dans le SCN. La mise en œuvre pratique des routines de calcul de la synergie, qui peut inclure des mesures de distance euclidienne, de similarité cosinus ou même des évaluations de compatibilité logique, sera explicitée à travers des exemples concrets et du pseudo-code. Ce chapitre servira ainsi de passerelle entre le concept théorique du DSL et sa traduction en algorithmes exécutables, en mettant particulièrement l'accent sur la cohabitation des entités symboliques et sub-symboliques au sein d'un réseau unifié.

Le **Chapitre 13** quant à lui se penchera sur la dimension plus globale et cognitive de l'approche, en explorant la perspective d'une **IA forte** ou d'un raisonnement cognitif avancé reposant sur une fusion neuro-symbolique. Ici, l'objectif sera d'étudier comment le SCN peut non seulement

organiser de manière auto-adaptative des représentations sub-symboliques mais aussi, de façon dynamique, pondérer et moduler des ensembles de règles ou des concepts logiques. Ce chapitre examinera, par exemple, comment la plasticité des pondérations  $\omega_{i,j}$  peut servir à ajuster l'importance relative de certaines règles en fonction de leur confirmation par des données empiriques. Nous discuterons également de l'**ordonnancement** et de la **mise à jour** des blocs de règles en interaction avec les embeddings issus de données perceptuelles, et nous montrerons comment ces mécanismes peuvent contribuer à une meilleure cohérence globale du système. L'approche neuro-symbolique proposée vise à dépasser la dichotomie classique entre réseaux de neurones et systèmes experts, en intégrant ces deux dimensions dans une même architecture adaptative.

Pour résumer, ces chapitres futurs illustreront de manière concrète les points suivants, en référence aux concepts développés précédemment dans (2.5.2.1) et (2.5.2.2) :

- **Architecture et implémentation pratique (Chapitre 5)** : Nous détaillerons comment structurer les données  $\Omega$  et les pondérations  $\omega_{i,j}$ , ainsi que les algorithmes de mise à jour adaptés aux scénarios hybrides où coexistent des entités symboliques et sub-symboliques. Des aspects tels que l'inhibition, le recuit simulé et les stratégies de gestion des clusters seront abordés pour permettre une auto-organisation efficace dans des contextes hétérogènes.
- **Perspectives cognitives et neuro-symboliques (Chapitre 13)** : Nous examinerons comment un SCN peut servir de colonne vertébrale à une architecture d'**IA forte**, où la logique formelle et le raisonnement symbolique se combinent avec la flexibilité et la plasticité des représentations neuronales. L'objectif sera d'illustrer, par des études de cas et des analyses théoriques, comment la fusion des deux approches peut conduire à une intelligence plus intégrée, capable de s'adapter à des environnements complexes et évolutifs.

En conclusion, le passage du cadre théorique présenté dans le **Chapitre 2** vers les chapitres ultérieurs (notamment **Chapitre 5** et **Chapitre 13**) marque une transition essentielle, de la **modélisation conceptuelle** à la **mise en œuvre opérationnelle**. Cette transition permettra de concrétiser l'idée que le DSL n'est pas seulement une abstraction mathématique ou un ensemble de principes inspirés de la physique statistique et des neurosciences, mais qu'il constitue également un outil puissant pour l'ingénierie de systèmes adaptatifs. Ces systèmes seront capables de gérer la cohabitation d'entités logiques et sub-symboliques de manière harmonieuse, ouvrant ainsi la voie à des applications dans des domaines variés tels que la robotique, les réseaux sociaux évolutifs, l'écologie numérique, et bien d'autres. Les futurs chapitres détailleront ainsi les aspects pratiques de cette approche interdisciplinaire, en se référant explicitement aux travaux et concepts développés dans les sections précédentes, notamment (2.5.2.1) et (2.5.2.2).

### *2.5.3. Apport en Robotique et Contrôle Adaptatif*

Les principes du **DSL** (Deep Synergy Learning) — basés sur la dynamique adaptative de liaisons  $\{\omega_{i,j}\}$  et la formation de clusters en fonction de la **synergie** — ne se cantonnent pas aux systèmes algorithmiques ou cognitifs : ils trouvent également un **terrain d'application** dans la **robotique**. En particulier, la notion de **coordination sensorimotrice** peut être envisagée comme un **réseau**

d'entités (capteurs, actionneurs, modules décisionnels) dont les liens se renforcent ou s'affaiblissent selon la **complémentarité** et l'**efficacité** qu'ils offrent. La section (2.5.3) examine l'**apport** du DSL en robotique et en contrôle adaptatif. Nous commençons (2.5.3.1) par rapporter quelques **retours d'expériences** de laboratoires ayant exploré la synergie adaptative pour la **coordination sensorimotrice**.

### 2.5.3.1. Retours d'expériences dans des laboratoires : usage de la synergie adaptative pour la coordination sensorimotrice

Dans le domaine de la robotique sensorimotrice, où la coordination entre divers capteurs et effecteurs constitue un enjeu crucial pour l'adaptation en environnement dynamique, plusieurs laboratoires ont expérimenté des approches inspirées du DSL (Deep Synergy Learning). Ces études, que l'on retrouve en référence dans la section (2.5.3.1) et reliées aux concepts développés dans les sections antérieures telles que (2.5.2.1) et (2.5.2.2), mettent en lumière comment l'auto-organisation des pondérations  $\omega_{i,j}$  peut être utilisée pour réguler de manière adaptative la coordination sensorimotrice.

#### A. Contexte général en robotique sensorimotrice

Dans une configuration robotique typique, un robot est équipé d'un ensemble hétérogène de capteurs — caméras, LiDAR, gyroscopes, microphones, etc. — ainsi que d'effecteurs (moteurs de roues, bras manipulateurs, etc.). Historiquement, la coordination de ces dispositifs s'appuyait sur des schémas centralisés ou des architectures modulaires fixes, qui ne prenaient pas toujours en compte la variabilité environnementale. Dans ce contexte, le DSL propose de considérer chaque capteur ou module comme une entité  $\mathcal{E}_i$  dont la relation avec un autre module  $\mathcal{E}_j$  est quantifiée par une pondération  $\omega_{i,j}$ . Cette pondération évolue selon la règle de mise à jour

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i,j) - \tau \omega_{i,j}(t)],$$

décrite en détail dans les sections (2.3.1.1) et (2.3.1.2), et permet ainsi d'exploiter la **synergie adaptative** entre capteurs et effecteurs. Lorsque la synergie  $S(i,j)$  entre deux modules est élevée—par exemple, lorsque deux caméras fournissent des informations complémentaires pour l'estimation de la profondeur ou pour la détection d'obstacles—la pondération  $\omega_{i,j}$  se renforce, favorisant une fusion efficace des données. Inversement, si un capteur est perturbé ou fournit des informations moins fiables, son lien avec d'autres modules est progressivement atténué.

#### B. Exemples de laboratoires et de scénarios rapportés

##### (a) Coordination multi-capteurs (Laboratoire A)

Dans un laboratoire spécialisé en vision stéréoscopique, un SCN a été déployé pour coordonner plusieurs caméras disposées à différents angles ainsi qu'un système inertiel (IMU). Les expériences ont montré que lorsque la qualité de la caméra frontale se détériore (par exemple, à cause d'un éblouissement), le système réévalue dynamiquement les pondérations, renforçant le lien entre la

caméra latérale et l'IMU. Ainsi, la fusion visuelle se fait de manière adaptative, garantissant une estimation de la profondeur plus robuste. Ce mécanisme repose sur la même règle de mise à jour décrite en (2.3.1.1) et (2.5.3.1), où l'auto-organisation des  $\omega_{i,j}$  permet de moduler la contribution de chaque capteur.

### (b) Coordination bras manipulateur et retours tactiles (Laboratoire B)

Un autre laboratoire s'est intéressé à la manipulation d'objets par des robots équipés de capteurs tactiles. Dans ce contexte, la synergie entre les capteurs tactiles et le contrôle moteur du bras manipulateur a été mise en œuvre à l'aide d'un SCN. En cas de prise stable, la pondération entre le capteur tactile et le module de commande du bras augmente, tandis que des erreurs (glissement ou échec de préhension) entraînent une diminution de  $\omega_{tactile,moteur}$ . Ce processus adaptatif, qui permet d'auto-ajuster les stratégies de préhension, illustre bien comment la mise à jour des pondérations assure l'émergence de clusters de connexions robustes, comme évoqué dans les sections (2.5.3.1) et (2.5.2.2).

### (c) Coordination dans un essaim de robots (Laboratoire C)

Dans le domaine de la **swarm robotics**, chaque robot est considéré comme une entité  $\mathcal{E}_i$ . Un SCN a été utilisé pour modéliser la coopération entre les robots, où la pondération  $\omega_{i,j}$  reflète la capacité de deux robots à collaborer efficacement (par exemple, pour l'exploration ou le transport). Lorsqu'une coopération productive est détectée (par exemple, un échange réussi de signaux ou une mutualisation des ressources), les liens entre ces robots se renforcent. Inversement, des interférences ou des redondances conduisent à une diminution des pondérations. Cette approche décentralisée permet à l'essaim de s'organiser en clusters spécialisés sans qu'un contrôleur central ne soit nécessaire, ce qui renforce la flexibilité et la résilience du système.

## C. Bilan des expérimentations : Impact sur l'adaptativité et la robustesse

Les expériences rapportées dans les laboratoires montrent plusieurs points essentiels :

- **Adaptation en temps réel** : Grâce à la règle de mise à jour, telle que rappelée en (2.3.1.1), le SCN ajuste continuellement les pondérations  $\omega_{i,j}$  en fonction de la qualité perçue de la synergie  $S(i,j)$ . Ce mécanisme permet au système de se reconfigurer instantanément en réponse aux variations environnementales, qu'il s'agisse d'une dégradation du signal d'un capteur ou d'une nouvelle situation d'interaction entre modules.
- **Décentralisation** : L'approche DSL évite l'utilisation d'un contrôleur hyper-centralisé. Chaque entité, qu'elle soit un capteur, un effecteur ou un robot dans un essaim, ajuste localement ses liaisons. Cette autonomie locale favorise la formation d'un réseau d'interactions organiquement structuré, conformément aux principes développés en (2.5.2.1) et (2.5.2.2).
- **Robustesse face aux perturbations** : La capacité d'auto-organisation permet de réduire l'impact des défaillances locales. Par exemple, dans le laboratoire A, la diminution de la synergie d'un capteur dégradé se traduit par une baisse de sa pondération, limitant ainsi l'influence négative de ce capteur sur la fusion globale. De manière similaire, dans les

configurations en essaim (Laboratoire C), la redondance des liens garantit que la perte d'un robot ne perturbe pas l'organisation globale du réseau.

Ces observations, en écho aux concepts théoriques détaillés dans les sections (2.3.1) à (2.5.2.2), confirment l'intérêt d'une approche basée sur la **synergie adaptative** pour la coordination sensorimotrice en robotique.

### *Conclusion*

Les retours d'expériences dans divers laboratoires illustrent clairement que l'**usage** du DSL pour la coordination sensorimotrice présente de nombreux atouts :

- Une **adaptation en temps réel** des liaisons  $\omega_{i,j}$ , assurée par la mise à jour  $\omega_{i,j}(t + 1) = \omega_{i,j}(t) + \eta [S(i, j) - \tau \omega_{i,j}(t)]$  (cf. sections 2.3.1 et 2.5.3.1), permet de maintenir une fusion efficace des données issues de multiples capteurs.
- La **décentralisation** des décisions, où chaque entité ajuste localement ses connexions, évite une centralisation excessive et renforce la robustesse du système.
- La **robustesse** face aux perturbations, grâce à la capacité d'auto-organisation du SCN qui réoriente les synergies en fonction de la qualité des signaux, assure que même en présence de défaillances locales, le système conserve une performance satisfaisante.

Ces résultats expérimentaux confortent l'idée que la vision DSL, telle qu'exposée dans les sections précédentes (notamment 2.5.2.1 et 2.5.2.2), offre un cadre prometteur pour la coordination sensorimotrice en robotique. La suite de la discussion, dans la section (2.5.3.2) et dans des chapitres ultérieurs (comme le chapitre 9 pour l'apprentissage continu et le chapitre 11 pour la robustesse), étendra ces conclusions à des applications de robotique collaborative et de swarm robotics, démontrant ainsi la polyvalence et l'efficacité du DSL dans des contextes réels.

#### **2.5.3.2. Potentiel pour gérer des flottes de robots, chacun se reliant localement, formant un SCN global**

Dans le cadre de la robotique collaborative, le concept de **Deep Synergy Learning (DSL)** offre une perspective innovante pour orchestrer la coordination entre de nombreux agents autonomes. En effet, chaque robot, considéré comme une entité  $\mathcal{E}_i$  au sein du réseau, dispose de l'opportunité de modifier ses liaisons  $\omega_{i,j}$  de manière autonome, suivant une règle de mise à jour telle que

$$\omega_{i,j}(t + 1) = \omega_{i,j}(t) + \eta [S(i, j) - \tau \omega_{i,j}(t)],$$

où  $\eta > 0$  représente le **taux d'apprentissage** et  $\tau > 0$  le **coefficient de décroissance**. La **synergie**  $S(i, j)$  entre deux robots peut être définie à partir de critères variés, tels que la proximité géographique, la complémentarité des capteurs, ou encore la réussite des tâches coopératives. Cette approche, que l'on retrouve également dans la section **2.5.3.1** relative aux retours d'expériences en robotique sensorimotrice, permet à chaque robot d'ajuster localement ses connexions sans recourir à une supervision centrale.



Sur le plan théorique, l'architecture du SCN pour une flotte de robots repose sur la capacité de chaque agent à calculer sa **synergie locale** avec ses voisins immédiats, ce qui s'exprime par des mises à jour distribuées des pondérations  $\omega_{i,j}$ . Si la synergie est élevée entre deux robots, le poids associé se renforce, tandis qu'en cas de collaboration inefficace ou de divergence de mission, le poids se réduit. Ce mécanisme favorise l'émergence de **clusters** d'agents fortement connectés, illustrant une organisation auto-adaptative du réseau. Les aspects mathématiques de cette dynamique reposent sur des équations différentielles discrètes, et l'auto-organisation du système est comparable à une descente de gradient dans un espace de configuration défini par les pondérations, comme nous l'avons détaillé dans la section 2.5.1.1.

L'un des avantages majeurs de cette approche réside dans la **décentralisation** de la mise à jour. Chaque robot, en traitant localement l'information issue de ses interactions, contribue à la formation d'un **SCN global** sans intervention d'un contrôleur centralisé. Cette propriété est cruciale lorsque le nombre de robots est élevé, car elle permet d'éviter une surcharge computationnelle et favorise la **robustesse** du système. En effet, la défaillance d'un robot n'affecte que localement le réseau, et le système peut se réorganiser automatiquement pour compenser la perte, une caractéristique déjà évoquée dans la section 2.5.3.1.

Par ailleurs, cette méthode permet d'implémenter des mécanismes d'**apprentissage continu** et de **robustesse**, comme ceux détaillés dans les chapitres futurs, notamment le **Chapitre 9** sur l'apprentissage continu et le **Chapitre 11** sur la robustesse. En pratique, la synergie  $S(i,j)$  peut être ajustée en temps réel en fonction de la performance des collaborations, de la qualité des données de capteurs ou d'autres critères d'interaction, de sorte que les pondérations  $\omega_{i,j}$  évoluent de manière dynamique. Par exemple, si deux robots collaborent efficacement pour réaliser une tâche, leur pondération tend à converger vers une valeur élevée, ce qui renforce leur intégration dans un cluster. À l'inverse, une interaction moins productive entraîne une diminution de  $\omega_{i,j}$ , permettant ainsi au réseau de se réorganiser de façon adaptative.

L'architecture ainsi proposée se distingue par sa **scalabilité**. Chaque robot n'a besoin de gérer que ses interactions locales, souvent limitées à un voisinage restreint, ce qui réduit la complexité globale du système. De plus, en exploitant des algorithmes de mise à jour inspirés du **recuit simulé** et des **mécanismes d'inhibition** (voir sections 2.4.4 et 2.4.5.1), le SCN peut éviter de se figer dans un minimum local non optimal et favoriser l'émergence d'une organisation plus stable et cohérente. Ainsi, l'auto-organisation du réseau repose sur une dynamique où la **synergie locale** guide la formation de clusters qui, au niveau global, constituent une structure robuste et résiliente face aux variations de l'environnement ou aux perturbations.

Pour conclure, le **DSL** appliqué aux flottes de robots offre un cadre conceptuel et opérationnel permettant la gestion auto-adaptative de réseaux distribués. La mise à jour locale des pondérations, régulée par des mécanismes de **synergie**, d'**inhibition** et de **recuit**, conduit à l'émergence de clusters de coopération efficaces. Cette approche, qui se base sur les principes de la théorie des systèmes dynamiques et de l'apprentissage adaptatif, permet de coordonner des ensembles d'agents de manière décentralisée tout en garantissant une résilience et une adaptabilité indispensables dans des environnements complexes. Comme nous l'avons établi dans les sections 2.5.3.1 et 2.5.2.2, ainsi que dans les perspectives évoquées dans les chapitres futurs (**Chapitre 9** et **Chapitre 11**), cette méthode ouvre la voie à des applications robustes en robotique coopérative, tout en illustrant l'interdisciplinarité entre les approches mathématiques, physiques et biologiques dans la conception de systèmes intelligents.

#### 2.5.4. Comparaison avec d'Autres Paradigmes (RL, GNN, etc.)

Les sections précédentes (2.5.1 à 2.5.3) ont montré que le **DSL** (Deep Synergy Learning) se prête à des **convergences** multiples : apprentissage profond, IA symbolique, robotique adaptative. Pour clore ce panorama, la section 2.5.4 compare brièvement le DSL à d'autres paradigmes très en vogue en IA moderne, notamment les **Graph Neural Networks (GNN)** et les approches de **Reinforcement Learning (RL)** ou de **clustering**. Nous commencerons (2.5.4.1) par situer le DSL vis-à-vis des GNN, avant de voir (2.5.4.2) ce qui le distingue par rapport aux algorithmes de clustering ou aux méthodes de RL, puis de conclure (2.5.4.3) en renvoyant à des analyses plus approfondies dans les prochains chapitres 6, 7, 8.

##### 2.5.4.1. Où se situe le DSL vis-à-vis des Graph Neural Networks (GNN) ?

###### A. Rappel sur les GNN

Les **GNN** traitent un **graphe** d'entrée, où chaque nœud est associé à un **vecteur** (feature) et les arêtes peuvent porter un label ou un poids. On applique des “convolutions” ou des “aggregations” sur les voisins pour mettre à jour les **représentations** de chaque nœud. L'objectif est souvent de réaliser une **tâche** supervisée ou semi-supervisée (classification de nœuds, prédiction de liens, etc.). Les poids du réseau (couches GNN) sont **appris** par backpropagation, en prenant en entrée la structure du graphe et les features des nœuds.

Dans la plupart des GNN, la **topologie** du graphe est **fixée** (ou du moins pas totalement réinventée à chaque itération). On applique des couches de type GNN (GCN, GAT, etc.) pour extraire des embeddings plus riches de chaque nœud ou de l'ensemble du graphe.

###### B. DSL = évolution de la *matrice de liaison*

Le **DSL** fait **évoluer** la **matrice**  $\omega_{i,j}$  (pondérations d'un SCN) en fonction d'une **synergie**  $S(i,j)$ . La structure du graphe n'est donc pas statique : les liens (et leurs intensités) peuvent se créer, se renforcer ou disparaître au fil du temps, selon la dynamique auto-organisée.

Au lieu de se baser sur un graphe fixe et d'y appliquer des “pass” de convolution (comme en GNN), le DSL vise la **construction** ou la **réorganisation** même de ce graphe. Il ne repose pas obligatoirement sur un objectif supervisé ; il peut fonctionner dans une **logique** auto-organisée ou faiblement supervisée.

Dans l'ensemble, on peut relever plusieurs points communs entre les GNN et le DSL. D'abord, les deux approches manipulent des entités (nœuds) ainsi que leurs liaisons (arêtes ou pondérations). Par ailleurs, il est possible d'associer des *features* spécifiques à chaque entité (ou nœud), de sorte que le réseau intègre des informations additionnelles sur ces entités.

Sur le plan des différences majeures, la première tient à la **structure** : les GNN partent d'un graphe dont la topologie est définie au préalable, tandis que dans le DSL, on fait évoluer (voire créer ou supprimer) les connexions  $\omega_{i,j}$  au fil de l'exécution, selon la dynamique auto-organisée. La seconde différence concerne le **type d'apprentissage** : les GNN relèvent généralement d'un cadre supervisé (minimisation d'une perte via backpropagation), alors que le DSL met en œuvre des

prises à jour dites “locales”, le plus souvent sans supervision stricte. La troisième divergence réside dans l’**objectif** poursuivi : les GNN visent la réalisation d’une tâche précise (classification, régression, etc.), tandis que le DSL se consacre à l’auto-organisation d’un réseau d’entités en fonction de leurs synergies, sans impératif d’inférence ou de prédiction.

### Conclusion

Le **DSL** et les **GNN** partagent l’idée de **modéliser** un graphe, mais s’en distinguent par la **dynamique adaptative** des liens (pondérations  $\omega_{i,j}$ ) et la **nature** (souvent auto-organisée) du DSL. Là où un GNN suppose la structure essentiellement *donnée*, le DSL la *construit/transforme* en continu selon la synergie. On peut néanmoins imaginer des **approches hybrides** (un GNN pour la propagation de features + un DSL pour ajuster la topologie du graphe), suggérant une piste de recherche combinant le meilleur des deux mondes.

#### 2.5.4.2. Qu’apporte-t-il de distinct par rapport aux algorithmes de clustering traditionnels ou aux approches de renforcement ?

Le **Deep Synergy Learning (DSL)** se présente comme une approche qui auto-organise un réseau de pondérations  $\{\omega_{i,j}\}$  au fil d’une dynamique évolutive, en mettant à jour continuellement chaque connexion selon la règle

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i,j) - \tau \omega_{i,j}(t)],$$

où  $\eta > 0$  désigne le **taux d’apprentissage**,  $\tau > 0$  le **coefficient de décroissance**, et  $S(i,j)$  représente la **synergie** mesurant l’affinité locale entre les entités. Cette formulation, qui a été détaillée dans les sections précédentes telles que 2.5.1.1 et 2.5.3.1, contraste fortement avec les approches traditionnelles de clustering et de renforcement.

D’un côté, les **algorithmes de clustering** traditionnels comme **k-means** ou **DBSCAN** procèdent généralement en mode *batch*. Par exemple, dans k-means, chaque point de données est assigné au centroïde le plus proche, puis les centroïdes sont recalculés par minimisation de la somme des distances intra-cluster. Cette procédure itérative se poursuit jusqu’à convergence, et le nombre de clusters,  $k$ , est fixé a priori. Ces méthodes reposent sur une mesure de distance statique, ce qui aboutit à une partition fixe de l’ensemble des données. En revanche, dans le DSL, les pondérations  $\omega_{i,j}$  évoluent en temps réel en réponse à la synergie locale entre entités, sans imposer une structure de partition rigide. Ainsi, les clusters dans le DSL apparaissent de manière émergente et évolutive, permettant à une même entité de maintenir des connexions significatives avec plusieurs groupes au lieu d’être confinée à une unique partition déterminée à l’avance.

D’un autre côté, les approches de **Renforcement (RL)** se fondent sur un cadre dans lequel un agent, en interaction avec un environnement, reçoit un état  $s_t$ , choisit une action  $a_t$  et perçoit une récompense  $r_t$ . L’objectif est d’apprendre une politique  $\pi(a | s)$  qui maximise la récompense cumulative sur le long terme. Les algorithmes de RL, tels que le Q-learning ou les méthodes de policy gradient, se concentrent donc sur la prise de décision séquentielle guidée par un signal de récompense explicite. Dans le DSL, la mise à jour des pondérations est déterminée par la **synergie**  $S(i,j)$  locale et n’implique pas de signal de récompense global ni de choix d’actions dans un espace d’états. Par conséquent, le DSL s’inscrit dans une dynamique d’**auto-organisation** où chaque entité

ajuste ses connexions de façon autonome, en se basant sur des critères de compatibilité mesurés localement, plutôt que d'optimiser une fonction de récompense cumulative.

Les **avantages** du DSL résident dans sa **flexibilité** et sa **capacité adaptative**. En effet, la mise à jour continue des  $\omega_{i,j}$  permet au système de s'ajuster en temps réel aux variations des données, contrairement aux méthodes de clustering traditionnelles qui opèrent sur un ensemble de données fixe. De plus, le DSL ne se contente pas de partitionner l'espace des données ; il permet à une entité d'entretenir des liens multiples, reflétant ainsi la complexité des interactions dans des environnements réels. Du côté des approches de renforcement, le DSL offre une méthode décentralisée d'auto-organisation sans nécessiter une modélisation explicite des récompenses, ce qui peut simplifier l'implémentation dans des systèmes multi-agents où la coordination se fait par des interactions locales plutôt que par des signaux de récompense centralisés.

Cependant, cette **vision énergétique** et adaptative présente également des limites. Lorsque la synergie  $S(i, j)$  varie de manière significative au fil du temps ou dépend fortement des pondérations elles-mêmes, la fonction potentielle globale ne reste plus statique et l'interprétation comme une descente de gradient devient moins évidente. Par ailleurs, dans des environnements hautement dynamiques, les mises à jour continues peuvent conduire à des oscillations ou à une instabilité qui ne sont pas facilement expliquées par les méthodes de clustering classiques ou par les algorithmes de renforcement qui, eux, reposent sur des critères de performance cumulés. De plus, la fusion des deux paradigmes (clustering statique et apprentissage par renforcement) est envisageable, mais elle nécessite une conception hybride qui peut complexifier la mise en œuvre du système.

En conclusion, le **DSL** se distingue à la fois des algorithmes de clustering traditionnels et des approches de renforcement par sa capacité à produire une **auto-organisation évolutive** du réseau de pondérations. La dynamique continue de mise à jour, basée sur la synergie locale, permet de former des clusters adaptatifs sans imposer de partitions fixes, tout en se passant d'un cadre décisionnel centré sur la maximisation de récompenses cumulées. Cette approche représente une alternative robuste et flexible, particulièrement adaptée aux systèmes complexes et distribués où les interactions entre entités sont non statiques et évolutives. Nous verrons, dans les chapitres **6**, **7** et **8**, comment ces principes sont intégrés dans des architectures d'IA avancées, démontrant ainsi la complémentarité et la richesse du DSL par rapport aux méthodes classiques.

#### 2.5.4.3. Discussion brève avant de laisser place à l'analyse plus poussée dans les chapitres 6, 7, 8

Dans les sections **2.5.4.1** et **2.5.4.2**, nous avons établi des comparaisons détaillées entre le **DSL** (Deep Synergy Learning) et divers paradigmes traditionnels, tels que les **Graph Neural Networks (GNN)**, les méthodes classiques de **clustering** et les approches de **Renforcement (RL)**. Il apparaît clairement que le DSL occupe un **créneau distinct** puisqu'il se concentre sur la **dynamique adaptative** des connexions  $\omega_{i,j}$  qui se mettent à jour continuellement en fonction de la **synergie** locale, sans imposer de schéma de supervision tel que requis dans les GNN, ni de partition fixe comme dans k-means, et sans recourir à un signal de récompense global comme c'est le cas en RL. Cette caractéristique intrinsèque permet, en outre, de combiner le DSL avec ces paradigmes afin de renforcer leurs capacités respectives, par exemple en ajustant dynamiquement la structure d'un

graphe dans un GNN ou en apportant une dimension d’auto-organisation dans un cadre de RL multi-agents.

La suite de notre ouvrage se poursuivra avec des développements plus approfondis dans les chapitres **6**, **7** et **8**, qui visent à clarifier et à intégrer ces notions dans des systèmes d’intelligence artificielle complexes. Le **Chapitre 6**, intitulé « **Apprentissage Synergique Multi-Échelle** », traitera de la capacité du DSL à gérer plusieurs niveaux ou granularités d’entités, en mettant en lumière la manière dont la mise à jour des  $\omega_{i,j}$  peut être effectuée dans une logique multi-échelle. Dans ce contexte, certains rapprochements avec les GNN et d’autres structures hiérarchiques seront approfondis, permettant ainsi de comprendre comment une architecture en couches peut émerger de l’auto-organisation des liens.

Le **Chapitre 7**, sous le titre « **Algorithmes d’Optimisation et Méthodes d’Adaptation Dynamique** », présentera des techniques issues du recuit simulé, de la compétition inhibitrice (cf. section **2.4.4**) ainsi que des approches inspirées de la physique statistique (cf. section **2.4.5.1**) pour optimiser la configuration d’un SCN. Ce chapitre abordera également les questions de **complexité** et de **scalabilité** lorsque le nombre d’entités augmente, en proposant des solutions pour assurer la convergence et la robustesse du système dans un environnement dynamique.

Le **Chapitre 8**, intitulé « **DSL Multimodal : Fusion de la Vision, du Langage et des Sons** », se focalisera sur les aspects pratiques de la fusion de données issues de différentes modalités. Ici, le DSL sera confronté aux approches traditionnelles de clustering multimodal et de RL multimodal, et l’on examinera comment la **synergie adaptative** entre divers types de données (images, textes, signaux audio) peut être exploitée pour former des représentations communes, enrichissant ainsi les performances du système global.

En guise de **conclusion** de cette discussion préliminaire, il apparaît que la dynamique du DSL se distingue des autres méthodes par sa capacité à organiser un réseau de pondérations de manière **auto-évolutive**. Ainsi, tandis que les GNN reposent sur une structure de graphe relativement fixe et un apprentissage supervisé, le DSL met en œuvre une **mise à jour continue** des  $\omega_{i,j}$  dans un contexte non supervisé. De même, les méthodes de clustering traditionnelles segmentent un ensemble de données statique, et les approches de RL se concentrent sur l’optimisation séquentielle d’actions via une récompense cumulative, alors que le DSL privilégie l’**auto-organisation** des relations sans imposer de partition ni de politique d’actions prédéfinie.

Les chapitres **6**, **7** et **8** approfondiront ces **intégrations** et complémentarités, démontrant comment le SCN peut s’insérer dans des architectures d’IA plus larges et plus riches. Ce passage de la théorie à l’ingénierie concrète ouvre ainsi la voie à des systèmes d’IA capables de combiner de manière harmonieuse les paradigmes de **clustering**, de **renforcement** et de **modélisation graphique**, contribuant à l’émergence de solutions adaptatives et évolutives dans des environnements complexes.

### *2.5.5. Transition vers les Chapitres Suivants*

L’ensemble de la section 2.5 (Perspectives Historiques et Liens avec l’IA Moderne) a révélé comment le **DSL** (Deep Synergy Learning) s’inscrit dans la continuité de plusieurs **courants** de l’IA : l’apprentissage profond (2.5.1), la neuro-symbolique (2.5.2), la robotique sensorimotrice

(2.5.3), et la comparaison avec GNN/clustering/RL (2.5.4). Pour **conclure** ce chapitre 2, la section (2.5.5) annonce la **transition** vers les chapitres plus concrets de la suite — notamment le **Chapitre 3**, qui portera sur la **représentation concrète** des entités et la définition pratique de la **synergie**, ainsi que les **Chapitres 4 et 5**, où l’on traitera davantage la **dynamique d’auto-organisation** et l’**architecture** SCN dans un cadre d’ingénierie.

### 2.5.5.1. Le prochain chapitre (3) traitera de la représentation concrète des entités et de la façon de définir la fonction de synergie dans divers contextes (image, texte, capteurs)

Dans la suite de ce travail, nous nous attacherons à passer d’une conceptualisation théorique du **DSL** (Deep Synergy Learning) à une mise en œuvre plus opérationnelle, en particulier en ce qui concerne la représentation concrète des **entités**. Le **Chapitre 3**, qui figure dans la table des matières, abordera de manière détaillée la problématique de la formalisation de chaque entité  $\mathcal{E}_i$ . Nous examinerons notamment si ces entités doivent être considérées comme des **vecteurs**—par exemple, des embeddings issus d’un réseau de neurones convolutionnel (CNN) ou d’un Transformer appliqué à des textes—ou si elles peuvent également prendre la forme de structures symboliques, telles que des ensembles de règles ou des ontologies. Dans ce contexte, il sera essentiel d’identifier les **structures de données** appropriées pour stocker et manipuler ces représentations, qu’il s’agisse de tableaux multidimensionnels, de graphes ou d’autres structures adaptées à la nature et à la dimension des descripteurs.

Un point central de cette démarche réside dans la définition de la **fonction de synergie**  $S(i, j)$ . Il s’agira d’établir des formules qui quantifient la similarité ou la complémentarité entre deux entités, en fonction du contexte considéré. Par exemple, dans le cas d’images, on pourra adopter une mesure basée sur la **distance cosinus** ou la distance euclidienne entre des embeddings extraits par un CNN, alors que pour des textes, des mesures de **similarité sémantique** issues de modèles de langage pré-entraînés (tels que BERT) seront privilégiées. Pour des capteurs, la synergie pourra être définie par des fonctions mesurant la cohérence temporelle ou la corrélation des signaux, telles que  $S(i, j) = \text{corr}(\mathbf{x}_i, \mathbf{x}_j)$ . Dans tous ces cas, la mise en œuvre devra garantir que la dynamique de mise à jour des pondérations  $\omega_{i,j}$  du **SCN** (Synergistic Connection Network) soit adaptée à la nature des données, ce qui implique la prise en compte de paramètres tels que le taux d’apprentissage  $\eta$  et le coefficient de décroissance  $\tau$  dans la règle de mise à jour

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(i, j) - \tau \omega_{i,j}(t)].$$

Cette équation, qui sera étudiée en détail dans les chapitres ultérieurs, devra être adaptée pour prendre en compte la diversité des types d’entités présentes dans le réseau.

Nous verrons également comment intégrer la **fonction de synergie** dans un cadre multimodal, où il est nécessaire de fusionner des informations provenant de sources très hétérogènes. Par exemple, dans un scénario de fusion de données, les représentations d’images et de textes devront être combinées au sein du même SCN, ce qui impliquera de définir des mécanismes permettant de normaliser et de pondérer les différentes mesures de synergie selon leur échelle et leur pertinence relative. Ce défi technique est essentiel pour assurer que la dynamique du DSL reste cohérente et exploitable, même lorsque les entrées proviennent de domaines très différents.



Le **Chapitre 3** constituera ainsi une passerelle entre la théorie développée dans le Chapitre 2 et les applications pratiques, en présentant des *pseudo-codes* et des exemples concrets d'implémentation. Il expliquera comment les entités, qu'elles soient purement sub-symboliques ou de nature mixte (symbolique et sub-symbolique), peuvent être représentées par des structures de données adaptées et comment la fonction de synergie  $S(i, j)$  est définie pour piloter la dynamique d'auto-organisation des pondérations.

Par ailleurs, la suite de la transition est explicitée dans les sections **2.5.5.2** et **2.5.5.3**, qui détailleront respectivement les aspects de la dynamique d'auto-organisation du SCN dans des contextes appliqués et la cohabitation des idées théoriques avec des principes d'ingénierie. Le **Chapitre 3** sera suivi par les **Chapitres 4 et 5** qui s'orienteront vers la mise en œuvre effective des algorithmes et la validation de la dynamique auto-organisée dans des scénarios réalistes. Cette progression permettra de passer de la conceptualisation abstraite des mécanismes du DSL à une réalisation pratique, en mettant l'accent sur l'extraction des **features** et l'adaptation de la fonction de synergie aux caractéristiques des données, qu'elles soient issues de l'imagerie, du traitement de textes ou des flux sensoriels.

En conclusion, ce prochain chapitre se propose de formaliser la représentation concrète des entités ainsi que la définition précise de la fonction de synergie, ce qui est indispensable pour assurer que le SCN opère de manière optimale dans divers contextes d'application. Les résultats présentés dans ce chapitre jetteront les bases pour les développements techniques approfondis qui suivront dans les chapitres **4** et **5**, lesquels se consacreront à la dynamique d'auto-organisation et à l'implémentation logicielle du DSL dans des environnements réels, tels que la robotique sensorimotrice et les systèmes multimodaux.

Références internes : cette section s'inscrit dans la continuité des sections **2.5.5.2** et **2.5.5.3**, lesquelles orientent respectivement vers la dynamique d'auto-organisation et la transition vers les chapitres suivants.

#### **2.5.5.2. Les chapitres 4 et 5 aborderont la dynamique d'auto-organisation plus appliquée, en tirant parti des bases mathématiques introduites ici**

Dans ce passage du Chapitre 2 aux chapitres suivants, nous avons posé des fondations théoriques solides pour le **DSL** (Deep Synergy Learning) en définissant précisément ses concepts clés, tels que la dynamique des pondérations  $\omega_{i,j}$ , les analogies avec la physique statistique, la théorie des systèmes dynamiques et les mécanismes d'**inhibition**. Cependant, il apparaît désormais nécessaire de transposer ces principes abstraits dans un cadre d'**implémentation concrète** afin de déployer le DSL dans des applications réelles. C'est à ce titre que les **Chapitres 4 et 5** interviendront, après que le **Chapitre 3** aura défini la représentation des entités (issues d'images, de textes, de capteurs, etc.) ainsi que la manière de formaliser la fonction de **synergie**  $S(i, j)$ .

Dans le **Chapitre 4**, intitulé "**Synergies Émergentes et Auto-Organisation**", l'objectif sera d'approfondir la manière dont la dynamique d'auto-organisation se manifeste au sein d'un SCN (Synergistic Connection Network). Nous passerons d'équations abstraites, telles que

$$\omega_{i,j}(t + 1) = \omega_{i,j}(t) + \eta [S(i, j) - \tau \omega_{i,j}(t)],$$

à des algorithmes précis et des schémas de mise en œuvre qui permettent de maîtriser et d'exploiter les synergies issues des interactions locales entre entités. La mise en œuvre algorithmique sera détaillée à travers des pseudo-codes qui illustreront la gestion du pas de temps, l'initialisation des pondérations et les stratégies de détection ainsi que de régulation d'oscillations (par exemple, via des mécanismes d'inhibition et de découpage en lots). Ces exemples concrets permettront de comprendre comment, itération après itération, des clusters émergents ou des micro-réseaux stables se forment naturellement à partir de mises à jour locales. Les concepts introduits dans les sections 2.3 et 2.4 – tels que la théorie des attracteurs, la fonction d'énergie potentielle définie en 2.4.3, et les mécanismes d'inhibition présentés en 2.4.4 – seront mobilisés pour expliquer en pratique comment ajuster les paramètres  $\eta$  et  $\tau$  pour obtenir une dynamique auto-organisée prévisible et ajustable.

Le **Chapitre 5**, intitulé "**Le Synergistic Connection Network (SCN) : Architecture Générale**", portera sur la conception complète du réseau. Ce chapitre s'attardera sur la construction de l'architecture du SCN en abordant les questions de stockage et d'organisation des pondérations  $\omega_{i,j}$ . Nous y discuterons des différentes structures de données susceptibles d'être utilisées (matrices denses versus structures creuses, listes d'adjacence, etc.), ainsi que de l'organisation hiérarchique du réseau, qui pourra être structurée en différentes couches ou niveaux de synergie (par exemple, un niveau local contrasté avec un niveau global). En outre, l'architecture du SCN sera décomposée en plusieurs modules fonctionnels, notamment le module de calcul de la synergie  $S(i, j)$ , le module de mise à jour des pondérations (intégrant les mécanismes d'inhibition et de saturation), le module de clustering pour extraire des groupes d'entités, et enfin une interface dédiée à l'ingestion de données en temps réel dans le cas d'un SCN évolutif. Des exemples d'implémentation seront fournis, avec des pseudo-codes illustrant un framework léger, susceptible d'être réalisé en Python ou en C++ couplé à CUDA pour des applications nécessitant un calcul parallèle.

Ces deux chapitres, 4 et 5, constituent ainsi un pont essentiel entre la théorie développée dans le Chapitre 2 et la phase d'**implémentation pratique** qui sera amorcée au **Chapitre 3**. La logique de progression consiste d'abord à définir concrètement la représentation des entités et la fonction de synergie dans le Chapitre 3, puis à explorer dans le Chapitre 4 la dynamique auto-organisée par les mises à jour locales, et enfin à concevoir une architecture complète pour le SCN dans le Chapitre 5. Cette transition vers l'ingénierie concrète permet de passer des principes mathématiques et théoriques à une réalisation opérationnelle, qui sera ultérieurement enrichie par des approfondissements sur le multi-échelle, l'optimisation avancée et l'intégration multimodale dans les chapitres 6, 7 et 8. Par ailleurs, les chapitres 9 et 10 aborderont l'apprentissage continu et le feedback coopératif, renforçant ainsi la robustesse globale du système.

### *Conclusion :*

Les **Chapitres 4** (synergies émergentes, auto-organisation) et **5** (architecture générale du SCN) constituent un pont fondamental dans le développement du DSL. Ils opérationnalisent les concepts abordés dans le Chapitre 2 en montrant comment implémenter et exploiter la dynamique adaptative des pondérations  $\omega_{i,j}$  dans des scénarios concrets. Cette transition permet également de préparer le terrain pour les chapitres ultérieurs, notamment les sections 2.5.5.3, 2.5.4.3 et les chapitres 6, 7, 8, qui exploreront l'intégration multi-échelle, l'optimisation et les applications multimodales du SCN. Pour plus de détails sur la transition entre théorie et pratique, les références internes aux sections 2.5.5.1 et 2.5.5.3 rappellent l'orientation initiale du Chapitre 2 et la synthèse des fondements



théoriques, tandis que les sections 2.4.5.2 et 2.5.4.3 présentent les perspectives d'intégration avec d'autres paradigmes.

### 2.5.5.3. Conclusion générale sur la place de ce chapitre 2 : on passe désormais d'un cadre historico-théorique à une mise en œuvre plus ingénierie pour initier le SCN

Ce chapitre 2 a permis de poser les fondements théoriques du **DSL** (Deep Synergy Learning) en établissant un panorama conceptuel riche et interdisciplinaire, qui relie des idées issues de la physique statistique, de la théorie des systèmes dynamiques et des modèles biologiques (tels que l'inhibition latérale et l'apprentissage Hebbien) aux approches contemporaines de l'IA et de l'auto-organisation. Les sections précédentes, notamment 2.5.1 qui aborde les précurseurs et l'évolution historique du DSL, 2.5.2 qui explore les potentialités de cohabitation entre entités symboliques et sub-symboliques, 2.5.3 qui présente des retours d'expériences dans le domaine de la robotique sensorimotrice, ainsi que 2.5.4 qui compare le DSL avec les paradigmes classiques (clustering, renforcement, GNN), ont permis de dresser un réseau de concepts centré sur la **synergie adaptative**. Nous avons vu que la dynamique des pondérations  $\omega_{i,j}$  peut être interprétée en termes d'énergie potentielle, de recuit simulé et de mécanismes d'inhibition, offrant ainsi une vision qui, bien que partielle dans les cas non stationnaires, constitue un cadre de référence solide.

En effet, la formulation d'une fonction d'énergie du type

$$\mathcal{J}(\mathbf{\Omega}) = - \sum_{i,j} \omega_{i,j} S(i,j) + \frac{\tau}{2} \sum_{i,j} \omega_{i,j}^2 + \dots$$

a permis d'illustrer comment, dans un scénario stationnaire, la mise à jour

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) - \eta \frac{\partial \mathcal{J}}{\partial \omega_{i,j}} \big|_{\omega_{i,j}(t)}$$

peut être assimilée à une **descente de gradient** locale, guidée par la synergie  $S(i,j)$  et régulée par le terme de décroissance  $\tau \omega_{i,j}$ . Ces concepts ont été détaillés dans les sections 2.4.3 et 2.4.4, qui exposent respectivement les aspects énergétiques et les mécanismes d'inhibition, et ils illustrent comment le DSL se distingue des méthodes statiques classiques.

Ce chapitre se positionne ainsi comme un **socle historico-théorique** en établissant d'une part l'origine et l'évolution des idées qui sous-tendent le DSL (voir 2.5.1 et 2.5.2) et, d'autre part, en comparant de manière critique le DSL avec d'autres paradigmes de l'IA, tels que les algorithmes de clustering traditionnels et les approches de renforcement présentées dans 2.5.4. Cette approche comparative montre que, contrairement aux modèles fixés ou purement supervisés, le DSL propose une **auto-organisation dynamique** des connexions  $\omega_{i,j}$ , qui s'adapte en continu aux évolutions des données et aux interactions entre entités.

La place du chapitre 2 dans l'ensemble de l'ouvrage est ainsi double. Premièrement, il offre une **vue d'ensemble** qui situe le DSL dans le contexte des développements historiques et théoriques de l'IA, en faisant le lien avec les principes de la **physique statistique**, de l'**auto-organisation** et des mécanismes **biologiques** (tels que l'inhibition latérale et la plasticité synaptique). Deuxièmement, il prépare le terrain pour une transition vers une **mise en œuvre ingénierie** plus concrète, où les

concepts théoriques seront traduits en structures de données, algorithmes et architectures logicielles. Cette transition est explicitée dans les sections **2.5.5.1** et **2.5.5.2**, qui annoncent respectivement la représentation concrète des entités et la dynamique d’auto-organisation, et qui servent de prélude aux chapitres suivants.

En effet, le **Chapitre 3** traitera de la **représentation et de la modélisation** des entités  $\mathcal{E}_i$  dans divers contextes (image, texte, capteurs), en définissant précisément la fonction de synergie  $S(i, j)$  adaptée à chaque type de données. Ensuite, les **Chapitres 4** et **5** se concentreront sur la mise en œuvre de la **dynamique d’auto-organisation** et la construction de l’**architecture globale** du SCN, en s’appuyant sur les principes établis ici – tels que la mise à jour des poids, les mécanismes d’inhibition, et les analogies avec des modèles physiques et biologiques. Enfin, les chapitres ultérieurs (notamment **6**, **7**, **8**, **9** et **10**) approfondiront les aspects d’**optimisation multi-échelle**, d’**apprentissage continu** et de **robustesse** du système.

### *Conclusion*

En somme, ce **Chapitre 2** se veut fondateur tant sur le plan **historico-théorique** que sur le plan **comparatif**, en établissant un réseau de concepts – tels que la fonction d’énergie potentielle, le recuit simulé, l’inhibition latérale et la plasticité adaptative – qui structure la dynamique des pondérations  $\omega_{i,j}$  dans un SCN. Il prépare ainsi le lecteur à une transition vers une phase d’**implémentation pratique**, où les idées théoriques seront traduites en algorithmes, en structures de données et en architectures logicielles concrètes. Cette démarche permet de passer du “pourquoi” au “comment” et d’initier l’exploitation effective du DSL dans des domaines variés, allant de la robotique sensorimotrice à l’IA multimodale, en passant par des applications en réseaux sociaux et en écologie numérique. Les références internes aux sections **2.5.1**, **2.5.2**, **2.5.3**, **2.5.4** et **2.5.5.1** illustrent la progression logique qui mène à cette transition, ouvrant la voie à une approche d’ingénierie concrète dans les chapitres suivants.