

# Introduction au Deep Synergy Learning

*Mohammed Bouayoun*

1.1. Contexte et Motivation

1.1.1. **Bref Historique de l'IA**

1.1.2. **Émergence des Réseaux Profonds (Deep Learning) et Limites**

1.1.3. **Des Approches Conventionnelles à la Synergie Informationnelle**

1.1.4. **Positionnement du DSL dans le Panorama de l'IA**

1.1.5. **Rôle Potentiel du DSL vers l'IA Forte (IAG)**

1.1.6. **Objectifs et Contributions Clés du DSL**

1.1.7. **Plan Général du Chapitre**

### 1.1.1. Bref Historique de l'IA

L'intelligence artificielle (IA) constitue un domaine de recherche qui s'inscrit dans une longue tradition, remontant à la première moitié du XXe siècle. Son évolution a été marquée par des jalons fondamentaux qui ont façonné la discipline telle qu'elle se présente aujourd'hui. Pour mieux appréhender l'émergence de l'Apprentissage Profond Synergétique (APS), ou **Deep Synergy Learning (DSL)** en anglais, il est essentiel d'examiner les principaux courants qui ont successivement vu le jour ou cohabité.

Les premières réflexions autour de l'intelligence artificielle trouvent leur origine dans les travaux d'Alan Turing, notamment à travers son article de 1950, *Computing Machinery and Intelligence*. Dans ce texte fondateur, il introduit le célèbre test de Turing, conçu pour évaluer la capacité d'une machine à simuler l'intelligence humaine dans le cadre d'une interaction textuelle.

C'est lors de la conférence de Dartmouth en 1956 que l'IA est officiellement définie comme un champ scientifique. Cette rencontre, organisée par John McCarthy, Marvin Minsky, Claude Shannon et Nathan Rochester, marque l'acte fondateur de la discipline. Elle établit les bases conceptuelles et méthodologiques nécessaires à la création de systèmes intelligents.

Les premières approches de l'IA reposaient sur la manipulation de symboles et de règles logiques pour modéliser la pensée humaine. Ces travaux, regroupés sous l'appellation **IA symbolique**, cherchaient à reproduire des processus cognitifs à travers des systèmes de règles formelles. Des programmes tels que *Logic Theorist* (1956) et *General Problem Solver* (1959) illustrent cette démarche en appliquant des algorithmes de recherche pour résoudre des problèmes mathématiques ou généraux.

Cependant, bien que performante dans des environnements fermés, l'IA symbolique montrait des limites lorsqu'il s'agissait de traiter des situations complexes ou incertaines. Cette incapacité à généraliser au-delà de scénarios spécifiques a révélé la nécessité d'approches complémentaires.

Parallèlement à l'IA symbolique, le courant connexionniste, basé sur les réseaux de neurones artificiels, gagne en popularité. Des chercheurs comme Frank Rosenblatt, Bernard Widrow et Marcian Hoff ont exploré le potentiel des réseaux d'unités élémentaires capables d'apprendre à partir de données, comme en témoigne le Perceptron (1958). Cependant, les travaux de Marvin Minsky et Seymour Papert, publiés en 1969, ont mis en lumière les limites des réseaux de neurones monocouches, incapables de traiter des problèmes non linéaires. Cette critique, combinée à l'absence de méthodes efficaces pour entraîner des réseaux multicouches, a conduit au déclin temporaire de l'intérêt pour le connexionnisme, marquant le premier "hiver de l'IA".

Le milieu des années 1980 voit la renaissance des réseaux de neurones grâce à la découverte de la rétropropagation, une méthode permettant d'entraîner efficacement des réseaux multicouches. Ce renouveau est accompagné d'une montée en puissance des approches statistiques de l'apprentissage automatique, telles que les modèles graphiques probabilistes et les machines à vecteurs de support (SVM). Ces outils élargissent les applications de l'IA à des domaines variés, notamment la vision par ordinateur et le traitement du langage naturel.

La fin des années 2000 marque l'avènement du Deep Learning, rendu possible par la convergence de trois facteurs majeurs. L'abondance des données massives issues d'Internet et des technologies numériques a offert un terrain fertile pour l'entraînement des modèles. L'émergence de ressources de calcul performantes, telles que les GPU et les TPU, a permis de traiter des volumes de données colossaux. Enfin, des progrès méthodologiques, incluant de nouvelles fonctions d'activation et des stratégies d'optimisation, ont renforcé l'efficacité des réseaux de neurones profonds. Ces avancées ont conduit à des percées significatives dans des domaines comme la reconnaissance d'images, la traduction automatique et les jeux stratégiques, illustrées notamment par le succès d'AlphaGo.

Malgré ces réussites, le Deep Learning présente certaines limites, notamment une forte dépendance aux données annotées, un manque d'interprétabilité et des difficultés à généraliser. Ces contraintes mettent en évidence la nécessité d'approches complémentaires.

Face aux limitations des méthodes actuelles, de nouveaux paradigmes ont émergé, mettant l'accent sur l'auto-organisation et les interactions dynamiques entre sources d'information. Le **Deep Synergy Learning (DSL)** s'inscrit dans cette dynamique en intégrant des notions de coopération et de coévolution au sein des systèmes intelligents. Cette approche vise à dépasser les modèles existants en favorisant une intelligence plus adaptable et autonome, ouvrant ainsi la voie à l'intelligence artificielle générale (IAG).

### Conclusion : un héritage pour le Deep Synergy Learning

L'évolution de l'IA, marquée par une succession d'approches symboliques, connexionnistes, statistiques et synergiques, illustre la quête permanente de systèmes plus performants et polyvalents. Le Deep Synergy Learning se propose de synthétiser et d'approfondir ces acquis, en mettant l'accent sur la synergie et l'adaptabilité dynamique, pour tendre vers une IA capable de rivaliser avec l'intelligence humaine.

### 1.1.2. Émergence des Réseaux Profonds (Deep Learning) et Limites

L'émergence des réseaux profonds, ou **Deep Learning**, constitue un tournant fondamental dans l'histoire de l'intelligence artificielle. Bien que les réseaux de neurones artificiels aient été introduits dès les années 1960, ce n'est qu'au début des années 2010 qu'ils se sont imposés comme une technologie centrale grâce à plusieurs avancées majeures. Ces progrès incluent l'accroissement massif des données numériques, le développement de ressources matérielles puissantes comme les GPU, et l'introduction de méthodologies innovantes. Cependant, malgré ces succès, le Deep Learning continue de présenter des limitations importantes qui suscitent l'exploration de nouvelles approches.

L'avènement du Deep Learning repose sur plusieurs éléments déterminants. D'une part, l'ère numérique a produit une quantité sans précédent de données issues de diverses sources, telles que des images, des vidéos, des textes ou des signaux. Ces ensembles de données volumineux, comme le dataset ImageNet, ont permis d'entraîner des réseaux plus profonds et plus complexes. D'autre part, l'utilisation des GPU, conçus initialement pour le traitement graphique, a permis d'accélérer les calculs massivement parallèles nécessaires à l'entraînement des modèles. Ces avancées matérielles ont été accompagnées de progrès méthodologiques, comme l'introduction de fonctions d'activation avancées, la régularisation par dropout ou encore la normalisation de lots (batch normalization). Enfin, les succès visibles dans des applications telles que la reconnaissance d'images, la traduction neuronale et les jeux stratégiques ont renforcé l'intérêt pour cette technologie.

Les réseaux de neurones profonds ont apporté des avantages significatifs dans le traitement des données complexes. L'une de leurs principales forces réside dans leur capacité à extraire automatiquement des caractéristiques pertinentes à partir des données brutes, ce qui simplifie considérablement les tâches d'ingénierie des features. De plus, leur performance dans des domaines tels que la classification d'images ou la reconnaissance vocale dépasse de loin celle des méthodes traditionnelles. Lorsqu'ils sont entraînés avec suffisamment de données pertinentes, ces réseaux montrent une certaine aptitude à généraliser en identifiant des motifs complexes que d'autres approches ne peuvent pas détecter.

Malgré leurs performances impressionnantes, les réseaux profonds présentent plusieurs limites. Tout d'abord, leur dépendance aux données annotées est problématique, car l'annotation de grands volumes de données est coûteuse et parfois impraticable dans certains contextes. Ensuite, leur nature de boîte noire rend difficile l'interprétation des décisions prises, ce qui pose des problèmes d'explicabilité, notamment dans des secteurs critiques comme la santé ou la justice. Les réseaux profonds sont également vulnérables aux attaques adversariales, où des perturbations mineures peuvent induire des erreurs majeures. De plus, leur capacité de généralisation reste limitée lorsqu'ils sont confrontés à des contextes différents de ceux rencontrés durant l'apprentissage. Enfin, le coût énergétique élevé de l'entraînement de ces modèles soulève des préoccupations en termes de durabilité et d'accessibilité.

Face à ces limitations, de nouvelles approches visent à repousser les frontières du Deep Learning. L'apprentissage peu supervisé ou auto-supervisé, qui réduit la nécessité de données annotées, est en plein essor. Par ailleurs, l'intégration du raisonnement symbolique avec les réseaux de neurones, dans une approche neuro-symbolique, pourrait permettre une intelligence plus transparente et explicable. Enfin, des recherches explorent des architectures auto-organisées où les informations interagissent de manière dynamique, rompant avec la rigidité des modèles actuels. Ces évolutions pourraient ouvrir la voie à des systèmes plus robustes, adaptatifs et proches d'une véritable intelligence artificielle générale.

### *1.1.3. Des Approches Conventionnelles à la Synergie Informationnelle*

L'histoire de l'intelligence artificielle témoigne de la succession de grandes tendances méthodologiques, chacune cherchant à reproduire ou modéliser certains aspects de l'intelligence humaine ou animale. Parmi ces approches, les approches symboliques et les approches statistiques ou connexionnistes ont dominé. Les premières s'appuient sur la formalisation explicite de la connaissance à travers des règles et des faits, tandis que les secondes reposent sur l'apprentissage à partir de grandes quantités de données via des modèles tels que les réseaux de neurones artificiels. Bien que ces paradigmes aient permis des avancées considérables, ils montrent leurs limites lorsqu'il s'agit de traiter des systèmes complexes, dynamiques ou émergents. La synergie informationnelle s'inscrit dans une tentative de dépasser ces restrictions en proposant une approche inspirée des systèmes naturels.

Les systèmes naturels, tels que les écosystèmes, les colonies d'insectes ou le cerveau humain, illustrent un principe fondamental selon lequel l'ensemble dépasse la somme de ses parties. Ces systèmes se caractérisent par l'auto-organisation, où des structures émergent sans planification globale, la robustesse grâce à la redondance des composants, et la capacité d'adaptation en réponse à des variations environnementales. Les entités locales interagissent dynamiquement à travers des signaux ou des mécanismes simples, conduisant à des comportements collectifs complexes. Transposé à l'intelligence artificielle, ce principe propose de considérer les informations comme des entités actives capables d'interagir, de coopérer et de s'adapter en permanence.

Les approches classiques du Deep Learning, bien qu'efficaces, restent enfermées dans une structure hiérarchique rigide où les données circulent de manière linéaire ou avec des boucles limitées. Cela restreint leur capacité à modéliser des interactions complexes ou des phénomènes émergents. En particulier, dans des scénarios multi-modaux combinant des flux hétérogènes comme la vision, le texte ou l'audio, l'intégration repose souvent sur des mécanismes de fusion statiques qui ne permettent pas une réelle interaction entre les modalités. Ces architectures peinent également à s'adapter à des environnements dynamiques ou partiellement inconnus.

La synergie informationnelle propose une vision radicalement différente. Les entités d'information, au lieu d'être des entrées passives dans un modèle prédéfini, deviennent des acteurs autonomes capables d'interagir et d'évoluer. Elles peuvent évaluer leur synergie avec d'autres entités, c'est-à-dire mesurer leur capacité à produire ensemble des informations plus riches que la somme de leurs contributions individuelles. En fonction de cette évaluation, elles peuvent nouer, rompre ou modifier des connexions, favorisant l'émergence de structures auto-organisées et dynamiques. Ce processus dépasse la simple adaptation des pondérations traditionnelles en permettant des changements topologiques dans les réseaux, où de nouvelles entités peuvent apparaître, fusionner ou disparaître.

Le **Deep Synergy Learning (DSL)** s'inscrit dans cette continuité en cherchant à unifier et enrichir les paradigmes existants. Alors que le Deep Learning met l'accent sur la profondeur des représentations, le DSL valorise la richesse et la dynamique des interactions entre les entités d'information. Les bénéfices attendus incluent une meilleure intégration des modalités multiples, une résilience accrue face à l'incertitude et une capacité d'apprentissage continu où le réseau peut évoluer en réponse à de nouvelles données sans nécessiter un



réapprentissage complet. En favorisant des interactions flexibles et adaptatives, le DSL ambitionne de jeter les bases d’une intelligence artificielle plus proche des capacités cognitives humaines.

#### 1.1.4. Positionnement du DSL dans le Panorama de l’IA

Le **Deep Synergy Learning (DSL)** s’inscrit dans la continuité des avancées en intelligence artificielle, tout en introduisant une approche novatrice fondée sur la coopération dynamique et l’auto-organisation des entités d’information. Pour mieux comprendre son originalité, il convient de le situer par rapport aux grandes tendances historiques et actuelles : l’IA symbolique, le connexionnisme incarné par le Deep Learning, et les approches hybrides ou émergentes.

Le DSL se distingue des systèmes classiques d’IA symbolique, qui reposent sur la formalisation explicite de règles logiques et de bases de connaissances, en permettant l’émergence spontanée de structures et d’organisations adaptées au contexte. Contrairement à la rigidité des approches symboliques, il s’appuie sur des mécanismes auto-adaptatifs qui modifient les interactions en fonction des données disponibles. Par ailleurs, il prolonge l’héritage connexionniste en mettant l’apprentissage au centre de son fonctionnement. Alors que le Deep Learning repose sur des architectures statiques et hiérarchiques comme les CNN ou Transformers, le DSL introduit une dynamique d’interaction entre les entités, permettant une réorganisation continue des connexions en fonction de leur synergie.

Le DSL puise son inspiration dans les systèmes naturels complexes tels que les écosystèmes ou les réseaux neuronaux biologiques. Ces systèmes se caractérisent par des propriétés globales qui émergent de l’interaction locale d’éléments simples, sans planification centrale. Dans cette perspective, chaque entité d’information dans le DSL peut établir, rompre ou renforcer des connexions en fonction de la synergie qu’elle partage avec d’autres entités. Cette dynamique dépasse les schémas hiérarchiques classiques en introduisant des boucles de rétroaction et en intégrant des structures internes évolutives. Contrairement aux approches traditionnelles où les flux d’information sont souvent indépendants et combinés à un stade tardif, le DSL favorise un tissage constant des flux, améliorant ainsi la robustesse et l’adaptabilité.

Le DSL se distingue également des approches neuro-symboliques qui intègrent explicitement des règles logiques dans des systèmes neuronaux. Bien qu’il ne repose pas directement sur des formalismes logiques, il permet l’émergence de règles implicites via des interactions coopératives et des clusters auto-organisés. De plus, il participe à la quête d’un apprentissage continu en permettant l’évolution permanente des connexions et en favorisant une réorganisation constante en réponse aux nouveaux contextes. Cela en fait une solution prometteuse pour surmonter le problème de l’oubli catastrophique souvent rencontré dans l’apprentissage automatique.

L’IA forte, ou intelligence générale, se caractérise par sa capacité à s’adapter de manière flexible à des environnements variés et non balisés. Le DSL offre une contribution unique à cet objectif en introduisant le principe de coévolution des entités d’information. Cette adaptabilité intrinsèque repose sur l’auto-organisation et la synergie, qui sont considérées comme des éléments essentiels pour l’émergence d’une intelligence créative et autonome. Plutôt que de remplacer le Deep Learning, le DSL peut s’y intégrer en tant que niveau supplémentaire, enrichissant les représentations neuronales classiques par une interactivité et une plasticité accrues.

### Conclusion

Le DSL occupe une place singulière dans le paysage de l’intelligence artificielle en combinant l’héritage des approches symboliques et connexionnistes avec les principes des systèmes complexes. Grâce à son orientation vers l’auto-organisation, la synergie et l’adaptabilité, il constitue une étape significative vers le développement d’une IA capable d’intégrer et de transcender les paradigmes actuels, tout en se rapprochant de la cognition humaine.

#### 1.1.5. Rôle Potentiel du DSL vers l’IA Forte (IAG)

L’**Intelligence Artificielle Forte (IAG)**, également désignée par intelligence artificielle générale, se définit comme la capacité d’une machine à réaliser des tâches cognitives humaines de manière autonome, flexible et adaptative, en s’appuyant sur une compréhension et un apprentissage comparable à ceux d’un être humain. Cet objectif ultime de l’intelligence artificielle reste encore largement théorique, malgré les avancées impressionnantes des systèmes actuels, notamment ceux basés sur le **Deep Learning**. Ces modèles, bien que performants dans des tâches spécialisées, peinent à atteindre le niveau de polyvalence et de plasticité requis pour prétendre à une véritable IA générale. Le **Deep Synergy Learning (DSL)** se positionne comme une approche novatrice et prometteuse, offrant un cadre conceptuel pour dépasser les limites des modèles traditionnels.

Une intelligence générale ne se limite pas à exceller dans une tâche unique, mais doit pouvoir évoluer dans des environnements variés et imprévisibles. Le DSL se distingue par son aptitude à favoriser l’apprentissage multi-contextuel grâce à ses mécanismes d’**auto-organisation** et de **synergie informationnelle**. Ces mécanismes permettent au système de s’adapter en temps réel à des situations inédites, en reconfigurant les relations entre entités d’information. Contrairement aux systèmes conventionnels, qui restent figés dans leurs architectures ou nécessitent une ingénierie lourde pour évoluer, le DSL peut ajuster sa structure de manière dynamique.

La richesse des interactions entre les entités est une autre caractéristique essentielle du DSL. Plutôt que de juxtaposer des modules spécialisés sans réelle interaction, il propose une approche où chaque flux d’information, qu’il soit visuel, textuel ou sensoriel, interagit et

co-évolue avec les autres. Ce type d'interconnexion fluide permet l'émergence de représentations conceptuelles globales, essentielles pour traiter la complexité du monde réel.

Enfin, le DSL intègre une dimension d'**adaptabilité évolutive**, permettant au système de non seulement apprendre, mais également désapprendre, réviser et transformer ses structures en fonction de nouvelles expériences. Cette adaptabilité dépasse celle des architectures classiques, qui, bien qu'elles ajustent leurs pondérations, conservent généralement une topologie statique.

L'un des principes fondamentaux du DSL est l'**auto-organisation**, inspirée des systèmes biologiques. À l'image du cerveau humain, il privilégie une architecture ouverte, où les entités d'information ne suivent pas une trajectoire linéaire prédéfinie, mais interagissent selon des schémas dynamiques et adaptatifs. Cette capacité à reconfigurer les flux internes permet au système de s'auto-réguler et d'évoluer en fonction des besoins contextuels.

La **plasticité dynamique** du DSL constitue un autre levier crucial pour atteindre l'IA forte. Elle dépasse l'ajustement traditionnel des pondérations en introduisant la possibilité de créer de nouvelles connexions, d'en supprimer d'obsolètes ou de réorganiser des clusters entiers. Cette plasticité ouvre la voie à une exploration continue et à une innovation constante, des caractéristiques indispensables pour développer une intelligence véritablement générale.

Les systèmes d'apprentissage profond ont démontré leur capacité à exceller dans des domaines spécifiques, comme la reconnaissance d'images ou la traduction automatique. Cependant, cette spécialisation s'accompagne d'une incapacité à transférer efficacement les connaissances d'un domaine à un autre. Le DSL propose une alternative en favorisant une **coopération dynamique** entre différentes modalités ou tâches, permettant une synergie globale entre sous-systèmes.

Ce paradigme ouvre la possibilité d'un transfert de connaissances fluide, où les acquis d'une modalité, comme l'analyse visuelle, enrichissent l'apprentissage dans une autre, telle que la compréhension du langage. Cette capacité de réutilisation et de réorganisation constitue une avancée majeure vers une **flexibilité généralisée**, essentielle pour le développement de l'IAG.

Le cerveau humain se distingue par son haut degré d'interconnexion et sa capacité à faire émerger des significations à partir de l'interaction entre différentes aires sensorielles et associatives. Le DSL s'inspire de ce modèle en proposant une architecture où les interactions dynamiques entre les entités d'information jouent un rôle central. Cette approche favorise non seulement l'émergence de comportements complexes, mais également celle de **schémas cognitifs** riches et adaptatifs, proches de ceux observés dans les processus humains.

De plus, le DSL vise à dépasser la simple corrélation statistique en permettant l'émergence de concepts et de significations. Les interactions entre entités d'information, structurées par des mécanismes d'auto-organisation, contribuent à la construction de **nœuds sémantiques**, amorçant ainsi une véritable compréhension du monde environnant.

Bien que le DSL représente une avancée prometteuse, plusieurs défis doivent être relevés pour qu'il puisse concrétiser son potentiel en tant que moteur de l'IAG. L'**efficacité algorithmique** constitue un premier enjeu, car les processus d'auto-organisation et de réorganisation dynamique peuvent être coûteux en termes de calcul. Trouver des solutions d'optimisation est donc essentiel.

L'**équilibre entre stabilité et plasticité** est également un aspect critique. Le système doit éviter de se réorganiser de manière excessive, ce qui pourrait nuire à la cohérence de ses apprentissages, tout en maintenant une capacité suffisante d'adaptation.

Enfin, les questions d'**interprétabilité** et de **sécurité** demeurent cruciales. Une architecture aussi dynamique et complexe pourrait poser des problèmes de traçabilité et de contrôle, notamment dans des domaines sensibles comme la santé ou la finance.

## Conclusion

Le **Deep Synergy Learning** se positionne comme une réponse innovante aux limites des approches traditionnelles de l'intelligence artificielle. En intégrant des principes d'auto-organisation, de plasticité dynamique et de coopération entre entités d'information, il ouvre la voie à une IA plus flexible, adaptative et contextuellement pertinente. Ces caractéristiques en font un candidat sérieux pour la réalisation de l'IAG, tout en posant de nouveaux défis techniques et éthiques qui devront être relevés pour concrétiser cette vision ambitieuse.

### 1.1.6. Objectifs et Contributions Clés du DSL

Le **Deep Synergy Learning (DSL)** s'inscrit comme une évolution fondamentale dans le domaine de l'**intelligence artificielle (IA)**, mettant en avant une **approche dynamique** et hautement **adaptative** pour surmonter les limites des architectures hiérarchiques classiques, telles que celles du **Deep Learning**. Cette nouvelle méthodologie repose sur l'idée centrale de **synergie informationnelle**, où les entités d'information interagissent et s'organisent de manière autonome pour produire des représentations plus riches et efficaces. L'objectif principal du DSL est d'intégrer les principes issus du **connexionnisme** et des **systèmes complexes**, tout en introduisant des mécanismes avancés d'**auto-organisation** et d'.

## A. Cadre théorique et objectifs fondamentaux

Le DSL repose sur une modélisation qui privilégie les interactions **synergiques** entre les entités d'information. Contrairement aux architectures fixes, où les flux d'information sont généralement dirigés et figés, le DSL permet une **reconfiguration continue** des liens internes. Cette flexibilité est rendue possible grâce à des mécanismes d'**évaluation dynamique** de la synergie, notée  $S(i, j)$ , entre deux entités  $i$  et  $j$ . Mathématiquement, la synergie entre deux entités peut être décrite comme une fonction dépendant de leurs représentations respectives  $x_i$  et  $x_j$  :

$$S(i, j) = \Phi(x_i, x_j),$$

où  $\Phi$  est une fonction d'évaluation qui peut être basée sur la corrélation, l'entropie conjointe, ou toute autre mesure adaptée au contexte.

L'objectif fondamental est de maximiser cette synergie globale dans le système, définie comme une somme pondérée des synergies individuelles :

$$\text{Synergie totale} = \sum_{i \neq j} w_{ij} S(i, j),$$

où  $w_{ij}$  représente les poids ajustables des connexions entre  $i$  et  $j$ . En optimisant cette fonction, le DSL favorise des **interactions pertinentes** et élimine les connexions inutiles ou redondantes, améliorant ainsi l'efficacité computationnelle et la robustesse des représentations générées.

L'un des objectifs clés du DSL est également d'offrir une plateforme pour l'**intégration multimodale**. En incorporant des données issues de différentes modalités, comme le texte, les images ou les signaux temporels, le système est capable d'unifier ces informations de manière **contextuelle**. Chaque modalité contribue activement à l'apprentissage global, tout en bénéficiant des enrichissements apportés par les autres, dans un cadre coopératif.

## B. Contributions principales et innovations méthodologiques

Le DSL propose une contribution notable en introduisant un mécanisme d'**auto-organisation** et de **plasticité structurelle**. Contrairement aux approches classiques où les architectures sont déterminées à l'avance, le DSL permet une **reconfiguration dynamique** des connexions entre entités, suivant des critères d'efficacité synergiques. Cette adaptabilité repose sur un processus d'optimisation continue où chaque entité ajuste ses liens en fonction de la pertinence des informations échangées. Ce mécanisme peut être formalisé comme suit :

$$\frac{\partial w_{ij}}{\partial t} = \alpha \cdot \nabla S(i, j),$$

où  $\alpha$  est un facteur d'apprentissage, et  $\nabla S(i, j)$  désigne le gradient de la synergie entre  $i$  et  $j$ .

Une autre innovation majeure réside dans la gestion des **clusters auto-organisés**, qui sont des regroupements spontanés d'entités hautement synergiques. Ces clusters peuvent émerger, se fusionner ou se dissoudre en fonction des interactions observées, offrant une **modularité naturelle** au système. Cette capacité de création et de dissolution des clusters renforce la **résilience** et l'**évolutivité**, deux qualités cruciales pour les environnements complexes et dynamiques.

Enfin, le DSL se distingue par sa capacité à capturer des **représentations sémantiques profondes**, en favorisant l'émergence de  **patrons globaux** à partir d'interactions locales. Cette propriété rapproche le DSL des processus cognitifs humains, où la compréhension découle souvent d'une synthèse d'informations contextuelles et émergentes.

## C. Avantages du DSL

Le DSL présente des avantages significatifs par rapport aux paradigmes d'apprentissage traditionnels. L'un des principaux bénéfices est sa capacité à **maximiser la synergie informationnelle**, ce qui améliore la richesse et la robustesse des représentations apprises. Cette optimisation synergique permet au système d'exploiter pleinement la diversité des données, y compris dans des environnements multimodaux. Par ailleurs, la **plasticité dynamique** du DSL garantit une adaptabilité exceptionnelle face à des scénarios évolutifs ou imprévisibles, tout en réduisant les besoins en supervision humaine. En outre, son cadre auto-organisé favorise une meilleure **résilience** aux perturbations, grâce à la redondance et à la flexibilité des connexions internes.

## D. Limites et défis à surmonter

Malgré ses nombreux atouts, le DSL n'est pas exempt de défis. L'un des obstacles majeurs est la **complexité computationnelle** inhérente à l'évaluation continue des synergies et à la reconfiguration des connexions. Cette complexité pourrait limiter son application à grande échelle, en particulier dans des contextes où les ressources en calcul sont contraintes. De plus, la nature hautement dynamique et adaptative du DSL pose des **problèmes d'interprétabilité**, car il devient difficile de tracer ou d'expliquer précisément les décisions prises par le système. Enfin, le maintien d'un **équilibre entre stabilité et plasticité** est une question délicate : trop de plasticité peut entraîner une instabilité, tandis qu'une structure trop rigide pourrait nuire à l'efficacité globale.

## Conclusion

Le **Deep Synergy Learning** se présente comme une avancée majeure dans le domaine de l'IA, en introduisant des mécanismes d'interaction et d'adaptation qui rompent avec les approches classiques. Grâce à sa capacité à maximiser les synergies informationnelles et à favoriser une auto-organisation dynamique, il ouvre de nouvelles perspectives pour la création de systèmes intelligents, flexibles et robustes. Cependant, pour atteindre son plein potentiel, il sera nécessaire de surmonter les défis liés à sa complexité computationnelle et à son explicabilité, tout en explorant des applications pratiques dans des environnements réels.

### 1.1.7. Plan Général du Chapitre

La présente section a pour objectif de donner une vision d'ensemble de la **structure** et de la **logique** qui sous-tendent le Chapitre 1 dans son intégralité. Après avoir introduit le contexte et la motivation du Deep Synergy Learning (DSL) dans la section 1.1, nous poursuivrons en explorant les différents volets nécessaires pour comprendre et situer le DSL dans l'écosystème de l'intelligence artificielle.



## 1.2. Définitions et Concepts Préliminaires

- 1.2.1. Qu'est-ce qu'une "Entité d'Information" ?
- 1.2.2. Notion de "Synergie Informationnelle"
- 1.2.3. Différence entre Interaction, Synergie et Corrélation
- 1.2.4. Approche Hiérarchique vs Approche Auto-Organisée
- 1.2.5. Réseaux Neuronaux Traditionnels vs Réseaux Synergiques
- 1.2.6. Terminologies Employées dans le DSL
- 1.2.7. Exemples Illustratifs de la Synergie dans la Nature

Cette partie clarifie les notions de base liées au DSL : nous y définissons les entités d'information, expliquons la différence entre simple interaction et synergie, et présentons des exemples concrets pour illustrer la pertinence de cette approche.

## 1.3. Importance de l'Auto-Organisation

- 1.3.1. Inspirations Biologiques et Cognitives
- 1.3.2. Concepts Clés : Émergence, Auto-Régulation, Feedback
- 1.3.3. Comparaison avec les Méthodes d'Apprentissage Classiques
- 1.3.4. Le Rôle des Flux d'Information Multimodaux
- 1.3.5. Évolution Dynamique et Adaptation en Continu
- 1.3.6. Impacts sur la Robustesse et la Résilience des Modèles
- 1.3.7. Perspectives pour une Approche plus Globale de l'Apprentissage

Nous soulignerons ici pourquoi l'auto-organisation est cruciale pour le DSL, en nous inspirant notamment des systèmes biologiques et cognitifs. Nous verrons en quoi elle se distingue des méthodes d'apprentissage habituelles et comment elle améliore robustesse et résilience.

## 1.4. Architecture Générale du DSL

- 1.4.1. Principe de Base : Entités et Liens Synergiques
- 1.4.2. Présentation du Synergistic Connection Network (SCN)
- 1.4.3. Notion de Cluster et de Macro-Cluster
- 1.4.4. Fonctions de Synergie : Distance, Similarité et Co-Information
- 1.4.5. Pondérations Adaptatives et Évolution Temporelle
- 1.4.6. Interactions Directes et Indirectes
- 1.4.7. Cas Particuliers : Synergie binaire et n-aire

Cette section décrit la structure interne du DSL. Nous y découvrirons notamment le **Synergistic Connection Network (SCN)**, ainsi que les mécanismes de mise à jour et de formation des clusters d'entités d'information.

## 1.5. Pourquoi une Approche Synergique ?

- 1.5.1. Avantages par Rapport aux Réseaux Neuronaux Profonds
- 1.5.2. Gestion Naturelle de la Multi-modalité

- **1.5.3. Flexibilité vis-à-vis des Données Incomplètes ou Bruitées**
- **1.5.4. Potentiel d'Auto-Évolution et d'Adaptation Continue**
- **1.5.5. Réduction de la Dépendance à la Supervision Humaine**
- **1.5.6. Création de Représentations Riches et plus Interprétables**
- **1.5.7. Intégration de Dimensions Symboliques ou Cognitives**

Ici, nous discutons des bénéfices concrets du DSL par rapport aux approches traditionnelles. Nous verrons comment la synergie facilite la fusion de modalités, l'adaptation aux données bruitées et la création de représentations plus expressives.

## **1.6. Applications Pressenties et Domaines Impactés**

- **1.6.1. Vision Artificielle et Reconnaissance d'Objets Complexes**
- **1.6.2. Analyse Audio et Traitement du Langage Naturel**
- **1.6.3. Robotique et Systèmes Intelligents Évolutifs**
- **1.6.4. Recommandation Personnalisée et Systèmes de Décision**
- **1.6.5. Surveillance, Diagnostic Médical et Anomalies**
- **1.6.6. Planification et Optimisation dans l'Industrie 4.0**
- **1.6.7. Perspectives pour la Recherche Fondamentale en IA Forte**

Nous dresserons un panorama des domaines susceptibles de bénéficier du DSL, en soulignant pour chacun d'entre eux comment la synergie et l'auto-organisation peuvent apporter des solutions nouvelles ou plus performantes.

## **1.7. Défis, Contraintes et Ouvertures**

- **1.7.1. Complexité Computationnelle et Scalabilité**
- **1.7.2. Qualité et Disponibilité des Données**
- **1.7.3. Développement d'Algorithmes d'Optimisation Appropriés**
- **1.7.4. Contrôle et Stabilité des Processus Auto-Organisés**
- **1.7.5. Interprétabilité et Explicabilité pour l'Humain**
- **1.7.6. Considérations Éthiques et Réglementaires**
- **1.7.7. Comparaisons Expérimentales avec d'Autres Approches**

Malgré son potentiel, le DSL se heurte à plusieurs obstacles : complexité des calculs, difficultés de collecte de données, incertitudes éthiques, etc. Cette section présente une synthèse de ces problématiques ainsi que des pistes pour y faire face.

## **1.8. Positionnement du DSL dans l'Évolution de l'IA**

- **1.8.1. IA Symbolique vs IA Sub-symbolique : Intégration Potentielle**
- **1.8.2. DSL et Apprentissage Profond : Collaboration ou Substitution ?**
- **1.8.3. Approches Hybrides : DSL, RL (Reinforcement Learning) et Logique**
- **1.8.4. Rôle de la Mémoire et de l'Attention dans le DSL**
- **1.8.5. Tendances Futures : Vers une IA Forte, Consciente ?**
- **1.8.6. Effet Sur la Recherche Interdisciplinaire**

- **1.8.7. Exemple de Convergence : DSL & Neurosciences**

Nous reviendrons sur la relation du DSL avec les grands axes de l'IA (symbolique, connexionniste, hybride) et discuterons de sa place actuelle et future dans l'avancée de la discipline vers l'IA générale.

## **1.9. Méthodologie, Ressources et Outils**

- **1.9.1. Approche de Recherche : Théorique, Expérimentale, Hybride**
- **1.9.2. Bases de Données et Plates-formes de Test pour le DSL**
- **1.9.3. Frameworks de Développement : Python, C++, Bibliothèques spécialisées**
- **1.9.4. Environnements de Simulation et d'Évaluation**
- **1.9.5. Protocoles de Validation : Qualitatifs et Quantitatifs**
- **1.9.6. Collaboration et Partage de Ressources (Open Source)**
- **1.9.7. Gestion du Cycle de Vie d'un Projet DSL**

Cette partie expose les moyens pratiques pour mettre en œuvre et évaluer le DSL : choix des plateformes logicielles, types de données disponibles, protocoles de validation... Elle servira de guide méthodologique pour les projets s'appuyant sur la synergie informationnelle.

## **1.10. Structure du Livre et Lecture Conseillée**

- **1.10.1. Vue d'Ensemble des Chapitres Suivants**
- **1.10.2. Lien entre Chapitres et Cohérence Globale**
- **1.10.3. Logique Pédagogique : Progression des Concepts**
- **1.10.4. Conseils de Lecture en Fonction des Profils (Débutants, Experts)**
- **1.10.5. Renvois vers des Ressources Complémentaires**
- **1.10.6. Planification du Lecteur : Stratégie d'Étude**
- **1.10.7. Perspectives de Recherche à Long Terme**

Ici, nous présenterons la trame globale de l'ouvrage, en expliquant comment chaque chapitre s'articule pour apporter une compréhension complète du DSL, que l'on soit débutant ou spécialiste confirmé.

## **1.11. Conclusion du Chapitre**

- **1.11.1. Synthèse des Points Clés**
- **1.11.2. Contributions du Chapitre à la Compréhension du DSL**
- **1.11.3. Liaison avec le Chapitre 2 : Fondements Théoriques Avancés**
- **1.11.4. Points de Discussion et Questions Ouvertes**
- **1.11.5. Réflexion sur la Place du DSL dans l'IA Généralisée**
- **1.11.6. Mot de Fin et Transition vers la Suite**

Enfin, nous dresserons le bilan des notions abordées, mettrons en évidence les contributions majeures de ce premier chapitre, et préparerons la transition vers le chapitre suivant, qui approfondira les fondements théoriques et mathématiques du **Deep Synergy Learning**.

Ce **Plan Général du Chapitre 1** doit permettre au lecteur de repérer rapidement les différentes thématiques abordées et de comprendre la **progression logique** qui va du contexte initial de l'IA (sections 1.1 à 1.2) aux **concepts structurels** du DSL (sections 1.3 à 1.5), puis à leurs **applications** (1.6), défis (1.7), positionnement (1.8) et **méthodologies** (1.9), avant de conclure sur l'organisation globale de l'ouvrage (1.10) et de refermer le chapitre (1.11).

### 1.1. Contexte et Motivation

Cette section introduit les raisons d'être du Deep Synergy Learning. Elle part d'un historique de l'IA (1.1.1) pour montrer l'émergence du Deep Learning (1.1.2) et ses limites, puis explique la transition vers l'idée de synergie informationnelle (1.1.3). Le **positionnement** du DSL dans l'écosystème de l'IA (1.1.4), son **rôle potentiel** dans la quête d'une IA forte (1.1.5) et ses **objectifs et contributions clés** (1.1.6) y sont détaillés. La sous-section 1.1.7 conclut en résumant la structure globale de cette partie introductive.

### 1.2. Définitions et Concepts Préliminaires

Après avoir cerné les motivations du DSL, on aborde ici les **fondations conceptuelles**. L'accent est mis sur la définition des "entités d'information" (1.2.1), la notion de "synergie informationnelle" (1.2.2), et les différences entre interaction, synergie et corrélation (1.2.3). On compare également l'**approche hiérarchique** traditionnelle aux mécanismes **auto-organisés** (1.2.4), en distinguant les **réseaux neuronaux classiques** des **réseaux synergiques** (1.2.5). Un point terminologique (1.2.6) clarifie le vocabulaire clé, avant de donner quelques **exemples illustratifs** de synergie dans la nature (1.2.7) pour mieux ancrer ces concepts.

### 1.3. Importance de l'Auto-Organisation

Le DSL met fortement l'accent sur l'**auto-organisation** : cette section en précise la portée et les inspirations (1.3.1), soulignant des **concepts clés** comme l'émergence et l'auto-régulation (1.3.2). On y compare la démarche auto-organisée avec les méthodes d'apprentissage classiques (1.3.3), en insistant sur le rôle crucial des **flux multimodaux** (1.3.4). L'**évolution dynamique** (1.3.5) et l'impact sur la **robustesse** (1.3.6) figurent parmi les arguments forts pour justifier ce choix d'organisation. La section se termine sur une réflexion quant à l'adoption d'une approche plus globale de l'apprentissage (1.3.7).

### 1.4. Architecture Générale du DSL

On explore ensuite la **structure interne** du Deep Synergy Learning. On démarre par les **principes de base** (1.4.1), puis on décrit le **Synergistic Connection Network (SCN)** (1.4.2), pivot du DSL, mettant l'accent sur la **notion de cluster** (1.4.3). Les **fonctions de synergie** et les mesures de distance ou similarité (1.4.4) sont présentées, suivies des **pondérations adaptatives** (1.4.5) qui évoluent au fil du temps. On souligne ensuite la distinction entre **interactions directes et indirectes** (1.4.6), et on conclut en mentionnant les **cas particuliers** (1.4.7) comme la synergie binaire ou n-aire, ouvrant la voie à des extensions plus avancées.

### 1.5. Pourquoi une Approche Synergique ?

Cette section développe les avantages potentiels du DSL. Elle souligne par exemple sa plus-value par rapport aux **réseaux neuronaux profonds** (1.5.1), sa gestion naturelle de la **multi-modalité** (1.5.2), et la **flexibilité** qu'il apporte face aux données bruyantes ou incomplètes (1.5.3). Viennent ensuite des points sur l'**auto-évolution** (1.5.4), la réduction de la **dépendance à la supervision** (1.5.5), l'émergence de **représentations plus riches** (1.5.6) et, enfin, l'éventuelle intégration de **dimensions symboliques** (1.5.7).

### 1.6. Applications Pressenties et Domaines Impactés

Ici, on brosse un **tableau des champs d'application** où le DSL peut exceller. On commence par la **vision artificielle** et la **reconnaissance d'objets** (1.6.1), l'**analyse audio** et le **traitement du langage** (1.6.2), la **robotique** (1.6.3), les **systèmes de recommandation** (1.6.4), la **surveillance** et le **diagnostic médical** (1.6.5), puis on aborde les problématiques de **planification** et d'**optimisation** dans l'industrie (1.6.6). La section s'achève en évoquant les implications pour l'**IA forte** (1.6.7).

### 1.7. Défis, Contraintes et Ouvertures

Même si le DSL est porteur de promesses, il soulève nombre de **défis**. On évoque notamment la **complexité computationnelle** (1.7.1), la qualité des **données** (1.7.2), la nécessité de **nouveaux algorithmes** (1.7.3), et les questions de **stabilité** dans les processus auto-organisés (1.7.4). Des enjeux d'**interprétabilité** (1.7.5) et d'**éthique** (1.7.6) sont également soulevés, avant de proposer des pistes de comparaison expérimentale (1.7.7) avec d'autres approches.

### 1.8. Positionnement du DSL dans l'Évolution de l'IA

On approfondit ici la **place** du DSL dans la grande histoire de l'IA, soulignant les possibilités d'**intégration** entre IA symbolique et sub-symbolique (1.8.1), les relations entre DSL et **Deep Learning** (1.8.2), et les perspectives d'**approches hybrides** (1.8.3). Le **rôle de la mémoire** et de l'**attention** (1.8.4), les pistes vers l'**IA forte et consciente** (1.8.5), ainsi que l'influence sur la **recherche interdisciplinaire** (1.8.6) sont passés en revue. Enfin, un **exemple de convergence** (1.8.7) montre comment le DSL peut s'enrichir des neurosciences.

### 1.9. Méthodologie, Ressources et Outils

Cette section vise à orienter le lecteur sur la façon de **mettre en œuvre** le DSL. On y aborde les approches de recherche (1.9.1), les **bases de données** disponibles (1.9.2) et les **frameworks** courants (1.9.3), ainsi que les environnements de **simulation** et d'**évaluation** (1.9.4).

Des **protocoles de validation** (1.9.5), la **collaboration open source** (1.9.6), et la **gestion de projet** (1.9.7) sont enfin exposés, pour accompagner toute personne souhaitant expérimenter avec le DSL.

#### 1.10. Structure du Livre et Lecture Conseillée

Avant de clore le chapitre, on donne ici un **aperçu global** de l'enchaînement des chapitres du livre (1.10.1), en expliquant la **cohérence** d'ensemble (1.10.2) et la **logique pédagogique** adoptée (1.10.3). On propose également des **conseils** selon les profils de lecteurs (1.10.4), indique des **ressources complémentaires** (1.10.5) et suggère une **stratégie d'étude** (1.10.6). La section se termine par quelques **perspectives de recherche** (1.10.7).

#### 1.11. Conclusion du Chapitre

Enfin, la conclusion du chapitre (1.11) dresse une **synthèse** (1.11.1) des points essentiels et met en évidence la **contribution** (1.11.2) de ce chapitre à la compréhension du DSL. On y établit une **liaison** claire avec le chapitre suivant (1.11.3), en soulignant les **questions ouvertes** (1.11.4) et les **réflexions** (1.11.5) sur la place du DSL dans l'IA généralisée. Un **mot de fin** (1.11.6) assure la transition vers la suite du livre, où seront approfondis les fondements théoriques et les premières implémentations concrètes.



## 1.2. Définitions et Concepts Préliminaires

Dans cette partie, nous allons introduire les notions fondamentales qui sous-tendent le **Deep Synergy Learning (DSL)**. Alors que la section précédente (1.1) portait sur le contexte historique, les motivations et la place du DSL dans l'écosystème de l'IA, le présent segment (1.2) est consacré à la **structure conceptuelle** du DSL et aux **fondements** qui permettent d'en comprendre les mécanismes profonds.

Nous débuterons par la définition de ce qu'est une **"Entité d'Information"** (1.2.1), pierre angulaire du paradigme DSL. Nous poursuivrons en expliquant le concept de **"Synergie Informationnelle"** (1.2.2) et en clarifiant la différence entre interaction, synergie et simple corrélation (1.2.3). Nous verrons ensuite comment le DSL se démarque d'une **approche strictement hiérarchique** (1.2.4) et en quoi il diverge des **réseaux neuronaux traditionnels** (1.2.5). Enfin, nous préciserons la **terminologie** spécifique (1.2.6) et illustrerons les principes de synergie par des **exemples concrets** (1.2.7).

### 1.2.1. Qu'est-ce qu'une "Entité d'Information" ?

Le concept d'**entité d'information** constitue l'un des piliers fondamentaux du **Deep Synergy Learning**. Dans le DSL, une entité d'information n'est pas un simple point de données isolé, mais plutôt un "objet d'apprentissage" possédant :

1. Une **représentation interne** (généralement un vecteur, un tenseur ou une fonction).
2. Des **caractéristiques dynamiques** : l'entité peut évoluer, se modifier, et entretenir des liens synergiques avec d'autres entités.
3. Un **historique** (ou une mémoire partielle) de ses interactions antérieures, pouvant impacter son comportement futur.

Pour formaliser une entité d'information  $\mathcal{E}$ , on se place dans un espace vectoriel (ou parfois un espace de Hilbert plus général) :

$$\mathcal{E}_k \in \mathbb{R}^d, \quad \text{ou} \quad \mathcal{E}_k \in \mathbb{R}^{n_1 \times n_2 \times \dots \times n_p},$$

suivant la nature des données (vecteur, matrice, tenseur, etc.). Par exemple, une image peut être encodée en tant que tenseur 3D (hauteur  $\times$  largeur  $\times$  canaux de couleur), tandis qu'un signal audio pourra être représenté sous forme de séries temporelles dans  $\mathbb{R}^d$ .

Dans certains cas, la représentation peut également être probabiliste. Ainsi,  $\mathcal{E}_k$  peut être décrite par une distribution de probabilité  $\mathcal{P}_k(\mathbf{x})$  sur un certain espace  $\mathbf{x} \in \mathcal{X}$ . L'important est de conserver la possibilité de **mesurer** la distance, la similarité ou la divergence entre deux entités :

$$\text{dist}(\mathcal{E}_k, \mathcal{E}_m) \quad \text{ou} \quad \text{sim}(\mathcal{E}_k, \mathcal{E}_m).$$

Outre la représentation brute, une entité peut avoir des **paramètres internes** (poids, biais, etc.) qui se modifient selon le temps ou selon les interactions :

$$\theta_k = \{\theta_{k,1}, \theta_{k,2}, \dots, \theta_{k,\ell}\}.$$

Ces paramètres influent sur le "comportement" de l'entité, c'est-à-dire sa manière de calculer des **scores de similarité** ou des **fonctions de sortie**. On peut aussi décrire un **état interne**  $\mathbf{s}_k(t)$  évoluant avec  $t$ , le temps (ou la phase d'apprentissage) :

$$\mathbf{s}_k(t) \in \mathbb{R}^d,$$

indiquant, par exemple, le niveau de confiance ou les caractéristiques discriminantes apprises jusqu'à l'instant  $t$ . Cet état peut servir de base à la mise à jour des **connexions synergiques** entre l'entité  $\mathcal{E}_k$  et d'autres entités  $\mathcal{E}_m$ .

Dans le cadre du DSL, nous pouvons associer à chaque entité  $\mathcal{E}_k$  un **ensemble** de composants :

$$\mathcal{E}_k = (\mathbf{x}_k, \mathbf{s}_k, \theta_k, \dots),$$

où :

- $\mathbf{x}_k \in \mathbb{R}^d$  est la représentation courante (ex. : vecteur de caractéristiques, image encodée).
- $\mathbf{s}_k$  est l'état interne dynamique (optionnel ou modulable).
- $\theta_k$  représente des **paramètres d'ajustement** ou d'apprentissage.
- D'autres composants pourraient inclure la **mémoire** (historique), les **métadonnées**, etc.

Cette formulation a pour but de **généraliser** la notion de "neurone" ou de "vecteur de données" pour en faire une entité d'apprentissage **active et adaptative**, au cœur des mécanismes d'interaction synergiques.

Pour illustrer concrètement la notion d'entité d'information, considérons la tâche de **reconnaissance de situations** dans une vidéo associée à un flux audio. Dans ce scénario, l'entité  $\mathcal{E}_{\text{visuelle}}$  peut représenter un **descripteur d'image** extrait par un réseau de neurones convolutionnel (CNN), enrichi de **paramètres** relatifs à la forme ou à la pose des objets détectés. De son côté, l'entité  $\mathcal{E}_{\text{auditive}}$  peut regrouper une **carte d'intensité fréquentielle** (spectrogramme) et un **état** décrivant la tonalité ou le niveau de bruit ambiant.

Ces deux entités ne constituent pas de simples “blocs” de données isolés : elles sont conçues pour **interagir, se synchroniser** ou même **fusionner**, dès lors que leur **synergie** (au sens de la section 1.2.2) est suffisamment élevée. Autrement dit, si les informations issues des canaux visuel et auditif s'enrichissent mutuellement, elles ont la possibilité de renforcer leurs liens et, potentiellement, de s'intégrer au point de former une entité commune, apte à traiter des signaux audiovisuels de manière coordonnée. Cette démarche souligne la **plasticité** du Deep Synergy Learning, qui autorise une reconfiguration permanente des relations entre entités pour améliorer la représentation globale de la scène.

En définitive, l'entité d'information constitue le **nœud élémentaire** du DSL. C'est à **travers** elle et **par** elle que les mécanismes de synergie prennent forme, permettant l'émergence de structures d'apprentissage plus complexes. Le **design** même de chaque entité, qu'il s'agisse de sa représentation (vecteur, tenseur, distribution de probabilité), de son état ou de ses paramètres d'ajustement, détermine directement l'**expressivité** et l'**efficacité** de l'apprentissage au sein du réseau. Le choix judicieux de ces attributs, adapté à la modalité (vision, audio, texte, etc.) et à la tâche visée, facilite la formation de **liaisons synergiques** fructueuses et, par conséquent, contribue à la robustesse et à la performance globale du système DSL.

### 1.2.2. Notion de “Synergie Informationnelle”

L'un des concepts centraux du Deep Synergy Learning (DSL) est la **synergie informationnelle**. Il s'agit de la capacité de deux (ou plusieurs) entités d'information à produire, ensemble, un **contenu** ou une **performance** impossible à atteindre (ou significativement moins bonne) si elles agissaient de manière isolée. Dans un cadre mathématique, la synergie se formalise par une mesure qui évalue l'**apport mutuel** entre les entités. Plus cette mesure est élevée, plus les entités concernées s'enrichissent mutuellement, amplifiant leur pouvoir de représentation ou de décision.

**Définition générale.** Considérons deux entités d'information,  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . En première approximation, on peut définir la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  comme une fonction qui quantifie à quel point la prise en compte conjointe de  $\mathcal{E}_i$  et de  $\mathcal{E}_j$  **améliore** un critère d'apprentissage (la prédiction d'une variable cible, la qualité d'une représentation, etc.).

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

où  $f(\cdot)$  peut être :

- une **mesure d'entropie conjointe** (ou de co-information) en théorie de l'information,
- un **gain de performance** par rapport à une référence (p. ex. différence de log-vraisemblance),
- une **fonction de similarité/distance** qui prend en compte des aspects non linéaires et adaptatifs.

Cette fonction  $f$  doit être conçue pour refléter la notion que “le tout est plus que la somme des parties”. Ainsi, il est d'usage de considérer qu'une **haute synergie** indique que l'association  $\{\mathcal{E}_i, \mathcal{E}_j\}$  est nettement plus informative que chacune des entités prise isolément.

En théorie de l'information, on peut s'appuyer sur l'**entropie conjointe** et la **co-information**. Par exemple, si  $\mathbf{X}_i$  et  $\mathbf{X}_j$  sont les variables aléatoires (représentant respectivement les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ ), on définit :

$$I(\mathbf{X}_i; \mathbf{X}_j) = H(\mathbf{X}_i) + H(\mathbf{X}_j) - H(\mathbf{X}_i, \mathbf{X}_j),$$

où  $H(\cdot)$  est l'entropie (de Shannon, ou d'autres formes d'entropie plus générales). Cette quantité  $I(\mathbf{X}_i; \mathbf{X}_j)$  mesure l'**information mutuelle** entre  $\mathbf{X}_i$  et  $\mathbf{X}_j$ . Toutefois, l'information mutuelle standard ne distingue pas toujours la **synergie** de la **redondance**.

Pour caractériser la synergie stricto sensu, plusieurs travaux de théorie de l'information proposent des mesures de **co-information** plus élaborées, voire des “Partial Information Decomposition” (PID), qui visent à séparer la part de redondance et la part de synergie :

$$\text{Synergie}(\mathbf{X}_i, \mathbf{X}_j) = I_{\text{PID}}^{(\text{syn})}(\mathbf{X}_i; \mathbf{X}_j \mid \mathbf{Y}),$$

où  $\mathbf{Y}$  peut être une troisième variable (cible à prédire) ou un contexte. Dans le cadre du DSL, il est donc pertinent d'utiliser, lorsque c'est possible, des **métriques entropiques** pour quantifier la contribution **non triviale** de chaque couple d'entités.

Une autre approche consiste à définir la synergie comme le **gain** (en termes de fonction objectif ou de performance) obtenu lorsqu'on associe deux entités, par rapport à leur utilisation séparée :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \Delta(\mathcal{E}_i, \mathcal{E}_j) = \Phi(\{\mathcal{E}_i, \mathcal{E}_j\}) - [\Phi(\{\mathcal{E}_i\}) + \Phi(\{\mathcal{E}_j\})],$$

où  $\Phi(\cdot)$  est un **score** ou une **mesure** de la qualité du système (ex. : taux de classification, log-likelihood, etc.). Dans ce cas :

- $S(\mathcal{E}_i, \mathcal{E}_j) > 0$  signifie qu'il y a véritablement une **valeur ajoutée** à combiner  $\mathcal{E}_i$  et  $\mathcal{E}_j$ .
- $S(\mathcal{E}_i, \mathcal{E}_j) < 0$  indique qu'il y a **inhibition** ou dégradation mutuelle.
- $S(\mathcal{E}_i, \mathcal{E}_j) = 0$  suggère une **indépendance** ou une simple addition sans synergie.

Cette formulation est souvent utilisée en pratique, car elle s'aligne directement sur un **objectif** (objectif supervisé, critère d'optimisation non supervisé, etc.). On peut de plus pondérer cette synergie par un facteur adaptatif, en tenant compte du **contexte temporel** ou **des autres entités** impliquées.

Contrairement aux approches linéaires ou statiques, le DSL prévoit que la synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$  soit **évolutive** au cours du temps. En d'autres termes, le réseau peut réévaluer en continu la contribution mutuelle de  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . Matériellement, cela se traduit par la mise à jour d'une **pondération synergiques**  $\omega_{i,j}(t)$  :

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

où  $\eta$  est un taux d'apprentissage,  $\tau$  un terme de régularisation (ou d'oubli). Plus la synergie entre deux entités est forte, plus leur lien s'intensifie. Au contraire, si ce lien n'apporte guère de valeur ajoutée (ou est carrément nuisible), sa pondération peut diminuer et aller jusqu'à **rompre** la connexion.

**Définition Synergie binaire, ternaire, et n-aire.** Dans sa version la plus simple, on considère la synergie entre **paires** d'entités ( $\mathcal{E}_i, \mathcal{E}_j$ ). Toutefois, nombre de scénarios exigent d'évaluer la synergie entre plusieurs entités simultanément. Dans ce cas, on généralise  $S$  à un ensemble  $\{\mathcal{E}_{k_1}, \dots, \mathcal{E}_{k_m}\}$ . On parle alors de **synergie n-aire**, dont la mesure n'est pas forcément la somme des synergies binaires. En effet, il se peut qu'une **triple**  $\{\mathcal{E}_a, \mathcal{E}_b, \mathcal{E}_c\}$  dégage une synergie supérieure à la somme des synergies de ses paires :

$$S(\mathcal{E}_a, \mathcal{E}_b, \mathcal{E}_c) > S(\mathcal{E}_a, \mathcal{E}_b) + S(\mathcal{E}_b, \mathcal{E}_c) + S(\mathcal{E}_a, \mathcal{E}_c).$$

Ce phénomène traduit la nature **émergente** du DSL, où des ensembles d'entités peuvent coopérer de manière non triviale pour engendrer de nouvelles représentations ou actions.

### Exemple scénario multimodal.

Soit

- $\mathcal{E}_{\text{visuelle}}$  (extraction de caractéristiques d'une image),
- $\mathcal{E}_{\text{auditives}}$  (traits de voix, intonation),
- $\mathcal{E}_{\text{textuelles}}$  (mots-clés extraits d'une transcription).

Si  $\mathcal{E}_{\text{visuelle}}$  et  $\mathcal{E}_{\text{textuelle}}$  ont peu d'information en commun, leur **information mutuelle** peut être faible. Pourtant, prises ensemble, elles peuvent produire un **contexte** (ex. : "lieu de la scène + thèmes abordés verbalement") qui aide à l'interprétation des sons (détection d'émotion). Autrement dit, c'est l'**intersection** de ces informations qui devient cruciale, expliquant une **synergie** plus forte lorsqu'on combine ces trois entités plutôt que deux à deux.

### Importance pour le DSL.

Le concept de synergie informationnelle est ce qui **différencie** le DSL d'un système où l'on se contenterait de propager les données entre couches. Au contraire, dans le DSL :

1. Les entités **cherchent** activement des partenaires synergiques,
2. Les **pondérations** entre elles **s'ajustent** en fonction de la synergie,
3. Les **clusters** ou micro-réseaux émergent autour des synergies les plus fortes,
4. Les entités peuvent **fusionner** ou **évoluer** pour mieux exploiter la coopération (nous verrons ces points dans les chapitres suivants).

Ainsi, la synergie agit comme un **moteur** d'auto-organisation et de **dynamique adaptative**, permettant au réseau de se **restructurer** au fil du temps, en valorisant les combinaisons d'entités les plus porteuses d'information ou de gain de performance.

### 1.2.3. Différence entre Interaction, Synergie et Corrélation

Lorsqu'on étudie les relations entre différentes entités d'information, il est essentiel de faire la distinction entre **interaction**, **corrélation** et **synergie**. Ces notions sont parfois utilisées de façon interchangeable, mais elles renvoient à des réalités mathématiques et conceptuelles différentes. Comprendre ces nuances permet de mieux cerner l'originalité du Deep Synergy Learning (DSL) et la portée de son concept de « synergie informationnelle ».

#### 1.2.3.1. Interaction : une relation générique

Le terme **interaction** désigne de manière générale l'influence mutuelle que peuvent exercer deux éléments (ou plus) l'un sur l'autre. D'un point de vue mathématique, on parle souvent d'**interaction** lorsque le comportement (ou la fonction) d'un système dépend de l'état de plusieurs variables de manière **non indépendante** :

$$f(\mathbf{x}_1, \mathbf{x}_2) \neq f_1(\mathbf{x}_1) + f_2(\mathbf{x}_2).$$

Par exemple, dans un modèle statistique de type régression, l'**effet d'interaction** entre deux variables se traduit par la présence d'un terme produit  $\mathbf{x}_1 \times \mathbf{x}_2$ .

Une interaction ne garantit pas nécessairement un effet bénéfique ou un « plus » collectif ; elle se borne à signaler que l'état ou la valeur prise par  $\mathbf{x}_2$  modifie l'effet de  $\mathbf{x}_1$  (et inversement).

Ainsi, dans le DSL, de simples **interactions** peuvent exister entre des entités d'information sans pour autant impliquer une **synergie** (cette dernière requérant un effet d'émergence véritable, voir plus bas).

#### 1.2.3.2. Corrélation : dépendance statistique (souvent linéaire)

La **corrélation** (au sens commun) mesure le degré de **dépendance statistique** entre deux variables, souvent réduit à la **corrélation linéaire** de Pearson :

$$\rho(\mathbf{X}, \mathbf{Y}) = \frac{\text{cov}(\mathbf{X}, \mathbf{Y})}{\sigma_{\mathbf{X}} \cdot \sigma_{\mathbf{Y}}}$$

avec  $\text{cov}(\cdot, \cdot)$  représente la covariance,  $\sigma_{\mathbf{X}}$  et  $\sigma_{\mathbf{Y}}$  désignent l'écart-type de  $\mathbf{X}$  et  $\mathbf{Y}$ .

Une corrélation élevée ( $\rho \approx 1$  ou  $\rho \approx -1$ ) signifie que deux variables évoluent de façon similaire (linéairement liée), tandis qu'une corrélation nulle ( $\rho \approx 0$ ) indique l'absence de dépendance linéaire (mais pas forcément l'absence de dépendance tout court).

**Remarque** : Dans un cadre non linéaire, d'autres mesures (mutual information, distance correlation, etc.) peuvent s'avérer plus pertinentes que la simple corrélation linéaire.

**Corrélation  $\neq$  synergie.**

Dans le cas d'une corrélation forte, on observe souvent une **redondance**. Si  $\mathbf{X}$  prédit bien  $\mathbf{Y}$ , alors  $\mathbf{Y}$  n'apporte pas nécessairement de nouvelle information. La corrélation peut aussi être trompeuse (corrélation de variables bruitées, effet de causalité inverse, variables cachées...).

Dans le DSL, deux entités très corrélées peuvent d'ailleurs être moins intéressantes (peu de gain) que deux entités faiblement corrélées, mais dont la **combinaison** génère un contenu supplémentaire.

Ainsi, une **corrélation forte** n'implique pas forcément une **synergie** ; et inversement, deux entités peuvent ne pas être corrélées mais créer, ensemble, un effet émergent.

La **synergie**, telle que définie dans le DSL, suppose un **gain** ou une **valeur ajoutée** lorsque les entités se combinent, au-delà de ce qu'elles apportent chacune de leur côté. Autrement dit :

$$S(\mathcal{E}_i, \mathcal{E}_j) > 0 \Rightarrow \text{La combinaison } \{\mathcal{E}_i, \mathcal{E}_j\} \text{ vaut plus que la somme séparée.}$$

- **Synergie  $\neq$  simple interaction** : L'interaction signale simplement une dépendance réciproque, alors que la synergie suppose qu'un nouveau niveau de fonctionnalité ou d'information émerge.

- **Synergie  $\neq$  redondance** : Deux variables très similaires (corrélées) ont peu de synergie, car prendre l'une ou l'autre n'ajoute pas grand-chose à la décision globale.
- **Synergie  $\neq$  coïncidence** : Les coïncidences peuvent être éphémères et non reproductibles. La synergie implique un **effet régulier** et **réel** sur l'optimisation ou la représentation interne du système.

Matériellement, dans un modèle où la fonction de coût  $\mathcal{L}$  est à minimiser (ou la fonction de performance  $\Phi$  à maximiser), la synergie entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$  s'exprime souvent comme :

$$\Delta_{ij} = \Phi(\{\mathcal{E}_i, \mathcal{E}_j\}) - [\Phi(\{\mathcal{E}_i\}) + \Phi(\{\mathcal{E}_j\})].$$

- Si  $\Delta_{ij} > 0$ , on parle de **synergie positive** (la combinaison est plus utile que la simple juxtaposition).
- Si  $\Delta_{ij} < 0$ , il y a **inhibition** ou **redondance néfaste**, et le couplage des entités se révèle contre-productif.
- Si  $\Delta_{ij} \approx 0$ , cela signifie qu'elles n'apportent pas grand-chose l'une à l'autre au regard de la tâche considérée.

Pour mieux illustrer ces différences, on peut imaginer un **diagramme** représentant trois situations :

1. **Corrélation (redondance)** : Les deux entités (A et B) apportent presque la même information.
2. **Interaction** : A et B se modifient mutuellement, mais sans forcément créer une nouvelle dimension.
3. **Synergie** : La combinaison A + B engendre un nouveau potentiel (ex. : l'ajout d'un flux audio + flux visuel crée un contexte multimodal plus riche que n'importe lequel des flux pris isolément).

#### Cas concrets.

- **Cas de redondance sans synergie** : Deux capteurs de température placés au même endroit, fournissant des mesures quasi identiques. Ils sont très corrélés, mais en prendre un seul est aussi informatif que d'en prendre deux.
- **Cas d'interaction sans synergie** : En biologie, certaines protéines interagissent (l'une bloque l'autre, par exemple), mais cela ne produit pas nécessairement un comportement globalement plus efficace pour l'organisme.
- **Cas de synergie forte** : En traitement de la parole, combiner la lecture labiale (analyse des mouvements des lèvres, entité visuelle) et le signal acoustique (entité auditive) améliore considérablement la reconnaissance par rapport à l'utilisation du signal audio seul ou de l'image seule, surtout en environnement bruyant.

Dans le **Deep Synergy Learning**, on cherche précisément à **favoriser** la création de synergies positives et à **réduire** (voire éliminer) les liens qui relèvent de la simple redondance ou d'interactions stériles. Les règles de mise à jour des **pondérations synergiques** (voir plus loin dans les chapitres dédiés) sont conçues pour :

1. **Renforcer** les liens entre entités ayant un  $\Delta_{ij} > 0$ .
2. **Diminuer** ou rompre les liens entre entités dont la corrélation ne procure aucun gain ou, pire, engendre un effet négatif ( $\Delta_{ij} < 0$ ).
3. Permettre la **détection** de synergies n-aires, où plusieurs entités coopèrent pour former des micro-réseaux auto-organisés.

Cette démarche permet d'éviter l'explosion combinatoire (en évaluant toutes les combinaisons) grâce à un **mécanisme dynamique** où les liens se forment ou se défont au fil de l'apprentissage, suivant les feedbacks de performance ou des indicateurs entropiques.

Dans le DSL, c'est précisément cette notion de synergie, mesurée et mise à jour en continu, qui permet de construire des **clusters** d'entités coopératives, de faire émerger de **nouvelles représentations**, et de potentialiser la **résilience** du système face à la variabilité des données. Les sections ultérieures reviendront sur la façon dont ce mécanisme de synergie se met en place dans une **approche auto-organisée** (1.2.4) et comment il se distingue des **réseaux neuronaux** traditionnels (1.2.5).

#### 1.2.4. Approche Hiérarchique vs Approche Auto-Organisée

Les approches hiérarchiques traditionnelles, largement utilisées dans le Deep Learning, reposent sur la succession de **couches** (layers) qui transforment progressivement les données d'entrée jusqu'à aboutir à une sortie (une prédiction, une classification, etc.). À chaque couche, on opère une **composition** de fonctions (le plus souvent linéaires, suivies de non-linéarités comme ReLU ou sigmoid). Par opposition, l'**approche auto-organisée** (telle qu'on la retrouve dans le Deep Synergy Learning, DSL) met l'accent sur la **capacité du**



**réseau à reconfigurer** ou **réorganiser** sa structure interne en fonction des synergies détectées, plutôt que de s'en tenir à une architecture rigide et prédéfinie.

Dans cette section, nous allons approfondir cette différence en examinant les principes fondamentaux de l'approche hiérarchique, pour mieux comprendre comment le DSL, en tant qu'approche auto-organisée, propose une alternative à la fois plus **dynamique** et plus **adaptative**.

### Fondements de l'approche hiérarchique

Dans un **réseau hiérarchique** traditionnel (tel qu'un réseau de neurones profond), le **principe du traitement en cascade** impose aux données  $\mathbf{x}$  de traverser une succession de transformations linéaires et non linéaires :

$$\mathbf{h}^{(1)} = f^{(1)}(\mathbf{x}), \quad \mathbf{h}^{(2)} = f^{(2)}(\mathbf{h}^{(1)}), \quad \dots, \quad \mathbf{h}^{(L)} = f^{(L)}(\mathbf{h}^{(L-1)}),$$

où  $\mathbf{h}^{(\ell)}$  représente la “représentation cachée” extraite à la couche  $\ell$ , et  $f^{(\ell)}$  désigne un opérateur paramétrique (incluant poids et fonction d'activation). Dans ce schéma, l'information **circule** essentiellement **de bas en haut**, et les éventuelles boucles de rétroaction (feedback top-down) demeurent limitées ou spécialisées, comme dans les architectures RNN ou LSTM.

Cette organisation induit une **séparation des rôles** :

- Les **premières couches** traitent des descripteurs “bas niveau” (par exemple, repérer des bords pour une image, des phonèmes pour un signal audio, etc.).
- Les **couches intermédiaires** approfondissent la combinaison de ces descripteurs, extrayant des motifs plus complexes.
- Les **dernières couches** produisent la **décision finale** (classe, score, etc.).

Toutefois, cette hiérarchie s'accompagne d'une **certaine rigidité**. Une fois le nombre de couches, la taille de chacune, et la nature des connexions (dense, convolutionnelle, récurrente...) choisis, la **structure** du réseau reste figée pendant la phase d'apprentissage. Seuls les **poids** sont ajustés par descente de gradient ou par l'une de ses variantes, tandis que la topologie globale demeure invariable.

Les **réseaux hiérarchiques** classiques présentent plusieurs limites marquantes. D'abord, ils demeurent fortement **dépendants** à la supervision : ils requièrent souvent de larges quantités de données annotées pour “régler” leurs poids internes. Par ailleurs, au fur et à mesure que l'architecture croît, on assiste à une **prolifération** exponentielle du nombre de paramètres, alourdissant le coût en calcul et en mémoire, tout en rendant le réseau plus difficile à **interpréter** et à **déboguer**.

En outre, ces réseaux manifestent un **manque d'adaptabilité** : face à un “domain shift” (changement de distribution des données), il s'avère nécessaire de procéder à un réapprentissage (ou un ajustement considérable), faute de mécanismes internes pour **reconfigurer** ou **réorganiser** dynamiquement la structure. Enfin, ils souffrent d'une **faible modularité** : bien que les couches s'empilent, elles ne peuvent guère “échanger” librement en dehors des cheminements établis dans l'architecture.

Dans une **approche auto-organisée**, caractéristique du DSL, la **plasticité topologique** occupe une place centrale : plutôt que de fonctionner avec des couches fixes, les **entités d'information** peuvent spontanément **former des clusters**, **créer** ou **rompre** des liaisons, et même **fuser** si la synergie s'avère suffisamment élevée (cf. sections 1.2.2 et 1.2.3). Ainsi, l'architecture n'est plus imposée : elle se construit et se reconstruit au fil de l'apprentissage, selon les besoins et l'évolution des données.

Cette **évolution dynamique** repose sur l'adaptation continue des pondérations synergiques  $\omega_{i,j}$  suivant une règle telle que

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

où  $\eta$  est un taux d'apprentissage,  $\tau$  un coefficient de régularisation et  $S(\mathcal{E}_i, \mathcal{E}_j)$  la synergie entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . Dans ce cadre, une synergie **positive** renforce la connexion, tandis qu'une synergie **négative** l'affaiblit. Ainsi, les entités dont la coopération est bénéfique sont encouragées à établir (ou consolider) leurs liens, tandis que les connexions moins productives s'éteignent naturellement.

Au fil de ce processus d'ajustement, des **micro-réseaux** ou **clusters auto-organisés** émergent dès lors que les synergies mutuelles s'élèvent entre certaines entités. Ces agrégats peuvent apparaître, se scinder ou disparaître, reflétant les évolutions des données ou l'arrivée d'interactions nouvelles. Par ailleurs, cette liberté structurelle favorise une **coopération multi-flux**, en autorisant les entités visuelles, auditives, textuelles, etc. à s'influencer **directement**, sans passer par un chemin prédéfini de “couches”. Par exemple, une entité auditive peut détecter à l'instant  $t$  une forte synergie avec une entité visuelle et, de ce fait, former un cluster pour la durée nécessaire ; plus tard, si les conditions changent, elle peut s'éloigner de ce groupe pour établir d'autres coopérations plus pertinentes.

Dans une **architecture auto-organisée**, on ne définit plus un enchaînement linéaire  $\mathbf{h}^{(1)} \rightarrow \mathbf{h}^{(2)} \rightarrow \dots$  ; à la place, on conçoit un **graphe**  $G(t)$  dont les **nœuds** sont les entités  $\{\mathcal{E}_1, \dots, \mathcal{E}_n\}$  et dont les **arêtes** représentent les pondérations synergiques  $\omega_{i,j}(t)$ . L'évolution du réseau se décrit alors par une fonction de mise à jour :

$$G(t+1) = \mathcal{U}[G(t), S(\cdot, \cdot)],$$

où  $\mathcal{U}$  tient compte des **critères de synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$ . Le système prend ainsi la forme d'un **Système Dynamique Non Linéaire**, se réorganisant de manière à privilégier les **combinatoires** d'entités jugées les plus utiles ou performantes.

Il est fréquent d'ajouter un mécanisme d'**énergie libre** ou de **coût global** :

$$\mathcal{J}(G) = - \sum_{i,j} \omega_{i,j} S(\mathcal{E}_i, \mathcal{E}_j) + \alpha \| \omega \| ^2,$$

de façon à **régulariser** la taille du réseau et à éviter que le nombre de connexions ne “flambe” de manière excessive. Dans ce cadre, la mise à jour peut s'effectuer via une **descente de gradient** (ou un algorithme d'optimisation inspiré des systèmes complexes, comme un algorithme génétique ou un recuit simulé), conduisant progressivement à une **organisation** qui valorise les synergies tout en restreignant les liaisons redondantes.

Dans une **approche auto-organisée**, l'**adaptabilité** représente un atout majeur : le réseau peut s'ajuster en continu face à l'arrivée de nouvelles données ou au changement d'une distribution, sans qu'il faille repenser entièrement son architecture. En outre, la **multi-modalité** y est gérée de façon native : les entités issues de différentes sources (audio, image, texte, etc.) ont la possibilité de **s'influencer** directement et de **former des clusters** multimodaux, ce qui facilite la **fusion** des divers flux et l'exploitation de leurs synergies. Par ailleurs, cette dynamique ouverte autorise un **potentiel créatif** : l'**émergence** de combinaisons inédites entre entités peut révéler des **patterns** jusque-là invisibles, que des architectures hiérarchiques classiques ne parviendraient pas à capturer aussi spontanément.

Néanmoins, cet avantage s'accompagne de plusieurs **défis** importants. D'abord, la **complexité de contrôle** peut s'avérer élevée : sans mécanismes de régulation (facteur  $\tau$ , pénalités, etc.), le réseau risque de basculer vers un excès de connexions ou de boucles menant à des oscillations. Le **coût de calcul** pose également question : évaluer la synergie entre un grand nombre d'entités exige souvent des heuristiques ou des méthodes parcimonieuses pour rester tractable en pratique. Enfin, l'**interprétabilité** peut devenir problématique : même si l'auto-organisation tends à faire émerger des clusters plus “significatifs”, l'évolution permanente de la structure complique l'analyse en profondeur du fonctionnement interne du réseau.

#### Comparaison synthétique

Caractéristique	Approche Hiérarchique	Approche Auto-Organisée (DSL)
<b>Architecture</b>	Fixe (définie a priori)	Flexible (graphe évolutif)
<b>Propagation de l'info</b>	Principalement feed-forward	Libre (coopération directe entre entités)
<b>Formation des connexions</b>	Statique (paramètres ajustés)	Dynamique (création / rupture de liens)
<b>Apprentissage</b>	Descente de gradient classique	Mise à jour des synergies (pondérations)
<b>Multimodalité</b>	Fusion tardive (généralement)	Intégration native, clusters multimodaux
<b>Adaptation continue</b>	Limité (fine-tuning, transfert)	Fort (reconfiguration à la volée)
<b>Exemples</b>	CNN, RNN, Transformers	Synergistic Connection Network (SCN)

Les sections suivantes (1.2.5 et suivantes) reviendront sur la comparaison plus directe entre les **réseaux neuronaux** traditionnels et les **réseaux synergiques**, tout en introduisant la terminologie spécifique (1.2.6) et des **exemples** (1.2.7) illustrant la pertinence de l'auto-organisation dans différents contextes naturels.

#### 1.2.5. Réseaux Neuronaux Traditionnels vs Réseaux Synergiques

Dans les sections précédentes, nous avons présenté les notions de **synergie informationnelle**, de **corrélation** et de **plasticité** structurelle dans une approche **auto-organisée**. Il est maintenant temps de faire un **parallèle** entre, d'une part, les **réseaux neuronaux profonds** (ou traditionnels) tels qu'on les connaît en apprentissage profond (Deep Learning) et, d'autre part, les **réseaux synergiques** comme envisagés dans le DSL (Deep Synergy Learning). Cette comparaison aidera à mettre en lumière ce que le DSL apporte de différent par rapport aux architectures classiques (CNN, RNN, Transformers, etc.).

Dans les **réseaux neuronaux traditionnels**, la structure est conçue dès le départ : on détermine à l'avance le nombre de couches et leur type (convolutionnelles, récurrentes, fully-connected, etc.), ainsi que la façon dont elles s'enchaînent. Chaque couche s'appuie sur la représentation produite par la précédente, imposant une **hiérarchie** explicite. L'apprentissage se fait en ajustant les poids et les biais via une descente de gradient ou l'une de ses variantes (Adam, RMSProp, etc.), tandis que la **propagation de l'information** suit principalement un chemin **feed-forward** ; même lorsque des boucles internes existent (RNN, LSTM), elles demeurent cantonnées à la topologie imposée.

À l'inverse, dans les **réseaux synergiques** (au sein du DSL), la topologie se veut **flexible et évolutive** : on ne parle plus de “couches” stricto sensu, mais d'un **ensemble d'entités d'information** liées entre elles par des pondérations **synergiques** (voir 1.2.2 et 1.2.4). Ces pondérations ne sont pas seulement ajustées ; elles peuvent aussi être créées, renforcées ou rompues, selon la synergie détectée. Cette possibilité de **reconfiguration** permanente marque la différence : le réseau ne se limite pas à empiler des couches, mais s'**auto-organise** en **clusters** lorsque la synergie l'exige. De plus, l'information ne circule pas selon une progression linéaire ; elle peut **transiter** entre toutes les entités jugées “synergiques”, adoptant ainsi une **approche distribuée** plus proche d'un écosystème vivant que d'un pipeline hiérarchisé.

Dans les **objectifs classiques** du Deep Learning, on minimise une **fonction de coût**  $\mathcal{L}(\theta)$  (par exemple, l'entropie croisée ou la MSE), à l'aide d'une **backpropagation** qui calcule les gradients. Les performances sont ensuite mesurées selon des métriques comme la **précision**, le **rappel** ou le **F1-score**, en fonction du type de tâche (classification, régression, etc.).

En revanche, dans une **approche DSL**, ces **objectifs traditionnels** (par exemple, la précision en classification) coexistent avec des **fonctions de synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  qui orientent l'apprentissage. Les mises à jour des pondérations synergiques  $\omega_{i,j}$  tiennent compte de ces scores de synergie, **favorisant** les liaisons dont la synergie s'avère significative, tout en permettant la **création**, le **renforcement** ou la **dissolution** de connexions. On peut également définir une **fonction globale**  $\mathcal{J}(G)$ , laquelle agrège la **somme** (ou autre forme d'agrégation) des synergies et **pénalise** la multiplication de connexions redondantes. Le réseau se comporte alors comme un **système dynamique**, visant à concilier la **minimisation** d'une perte liée aux tâches classiques (p. ex. une fonction de classification) et la **maximisation** de la synergie informationnelle au sens large.

Dans les **réseaux neuronaux traditionnels**, la **multimodalité** est gérée en concevant à l'avance des voies de traitement spécifiques (audio, image, texte, etc.) qui sont ensuite **fusionnées** à un stade donné : cela peut être au niveau de couches intermédiaires ou, plus souvent, par une fusion “tardive” vers la fin du pipeline. L'architecture doit donc être **explicitement conçue** pour chaque canal (par exemple, un CNN dédié à l'image, un RNN ou un Transformer pour le texte, puis un module spécialisé pour agréger les différents flux). La synergie potentielle entre ces canaux se découvre **indirectement**, via la backpropagation, mais la structure globale du réseau — et la manière dont les canaux se croisent — reste imposée de l'extérieur.

À l'inverse, dans les **réseaux synergiques** du DSL, les **entités** associées à divers canaux (audio, visuel, textuel, etc.) ont la possibilité de **se “découvrir” mutuellement** au fil de l'apprentissage. Si l'audio et l'image présentent une forte synergie, elles peuvent **former un cluster multimodal** de manière autonome, sans qu'une couche de fusion spécifique ne soit paramétrée au préalable. Cette approche facilite la **co-évolution** des représentations : si, à un moment donné, le flux visuel est perturbé par du bruit, l'entité visuelle peut s'appuyer davantage sur les canaux texte ou audio, à condition qu'une synergie élevée soit détectée. Ainsi, la multimodalité se développe de façon **organique**, guidée par la recherche de gains effectifs de performance ou d'information.

Dans les **réseaux neuronaux profonds** classiques, la **rétropropagation** constitue le mécanisme standard pour ajuster les **poids** de couche en couche ; on dispose alors d'une **architecture** explicitement définie de bout en bout et d'un **objectif** scalaire unique qui oriente la descente de gradient (p. ex. l'entropie croisée). En revanche, dans un **DSL**, les **mises à jour** des pondérations synergiques se réalisent de manière **distribuée**, souvent selon des règles plus locales (inspirées, par exemple, d'approches “Hebbiennes généralisées” ou d'une évaluation directe des gains de performance obtenus). Un **objectif global** peut persister (comme un taux de reconnaissance), mais la **découverte** de synergies s'opère fréquemment dans des configurations plus indépendantes du gradient global.

La **rétropropagation** n'est pas pour autant **exclue** : on peut envisager un **système hybride** où la backprop s'applique à certains sous-modules, tandis que la **formation** et la **reconfiguration** du graphe synergique suivent des lois d'auto-organisation distinctes. Ainsi, on bénéficie d'une **flexibilité** accrue : on continue à ajuster finement des parties du réseau via le gradient, mais on laisse aussi au réseau la possibilité de **découvrir** et de **renforcer** localement des liaisons synergiques au-delà du cadre strict imposé par une architecture fixe.

Dans les **réseaux neuronaux traditionnels**, la **robustesse** dépend essentiellement de la qualité du jeu d'entraînement et de plusieurs mécanismes de **régularisation** (dropout, batch normalization, etc.). L'**adaptation** à un nouveau domaine s'effectue souvent par un transfert d'apprentissage (transfer learning), suivi d'un **fine-tuning** partiel ou complet des poids du réseau. Cependant, ce procédé peut exposer le système au risque de **catastrophic forgetting**, lorsqu'on l'emploie pour apprendre de façon continue une succession de tâches : les poids ajustés pour les nouvelles données tendent alors à effacer ce qui avait été acquis auparavant.

Dans les **réseaux synergiques**, au contraire, la **structure évolutive** permet au système d'**allouer** de nouvelles entités ou de **renforcer** certains liens pour absorber plus facilement un **changement** de données, sans exiger un réapprentissage complet de l'ensemble du réseau. Les **clusters** déjà formés pour des tâches précédentes peuvent coexister dans la nouvelle configuration, au lieu d'être remplacés ou écrasés. Ainsi, un réseau synergique peut mieux **retenir** l'expérience passée (réduisant d'autant le **catastrophic forgetting**) et faire preuve d'une plus grande **flexibilité** quand son environnement ou sa mission évoluent.

Les **réseaux synergiques** n'ont pas vocation à **remplacer** purement et simplement les architectures neuronales traditionnelles. Au contraire, divers **scénarios de cohabitation** sont envisageables. On peut, par exemple, adopter une **approche hybride**, dans laquelle un pipeline CNN (pour l'image) ou Transformer (pour le texte) extrait des **représentations** initiales ; ces représentations deviennent ensuite des **entités** au sein d'un réseau synergique, lequel peut alors coopérer et se reconfigurer de manière plus libre.

Dans certains **systèmes complexes**, on peut aussi instaurer une **transition progressive**, en commençant par des couches de feature extraction classiques, puis en insérant une **couche synergique** à un stade où les divers canaux se croisent. De cette façon, on préserve la puissance des modèles traditionnels pour l'extraction de caractéristiques tout en intégrant la logique auto-organisée et adaptative du DSL à un niveau plus élevé.

Enfin, il est possible de développer des **extensions spécialisées**, par exemple un composant auto-organisé dédié à la **fusion multimodale** ou à la **gestion de multiples contextes**, tandis que la classification finale demeure assurée par un réseau fully-connected ordinaire. L'essentiel est d'exploiter la **flexibilité** des réseaux synergiques dans les domaines où ils excellent — par exemple, l'émergence dynamique de clusters — tout en s'appuyant sur l'expérience accumulée des architectures neuronales traditionnelles.

## Synthèse et perspectives

Aspect	Réseaux Neuronaux Traditionnels	Réseaux Synergiques (DSL)
<b>Topologie</b>	Fixe, pré-spécifiée	Évolutive, auto-organisée
<b>Propagation</b>	Hiérarchique, feed-forward	Dispersée, multidirectionnelle
<b>Apprentissage</b>	Backpropagation end-to-end	Règles locales + mise à jour synergie
<b>Évolution temporelle</b>	Nécessite du re-training pour s'adapter	Adaptation dynamique à la volée
<b>Gestion multimodale</b>	Fusion tardive ou intermédiaire, souvent manuelle	Fusion spontanée via synergie et création de clusters
<b>Robustesse</b>	Vulnérabilité à l'overfitting, besoin de régulariser	Auto-régulation via le feedback de synergie
<b>Applications</b>	Classification, régression, vision, NLP...	Idem, mais avec en plus la souplesse et l'auto-organisation

En conclusion, les **réseaux neuronaux traditionnels** et les **réseaux synergiques** diffèrent principalement par la **structure**, la **dynamique d'apprentissage** et la **capacité d'auto-organisation**. Le **Deep Synergy Learning** apporte une philosophie plus **organique**, inspirée des systèmes complexes, pour que l'intelligence artificielle puisse gérer la **variabilité**, la **multimodalité**, et s'auto-adapter en continu.

La section suivante (1.2.6) clarifiera la **terminologie** propre au DSL — notamment les notions de **clusters**, **entités**, **pondérations synergiques**, etc. — puis nous verrons (1.2.7) des **exemples illustratifs** tirés de la nature ou d'applications concrètes, afin de matérialiser les principes évoqués dans ce chapitre.

### 1.2.6. Terminologies Employées dans le DSL

Au fil des sections précédentes, plusieurs notions-clés sont apparues pour décrire les principes du **Deep Synergy Learning (DSL)**. Il est important de les clarifier et de les organiser en un vocabulaire cohérent, car ces termes forment la **boîte à outils conceptuelle** indispensable pour aborder les mécanismes internes et les applications pratiques du DSL. Dans cette section, nous passons en revue les principaux termes et leur signification, en soulignant les liens entre eux.

**Entité d'Information (ou “Information Entity”)** : Dans le DSL, une **entité d'information** (souvent notée  $\mathcal{E}_i$ ) représente l'unité fondamentale du système. Contrairement à un simple vecteur de données, une entité est un *objet d'apprentissage* pouvant inclure :

1. Une **représentation** (par ex. un vecteur, un tenseur, ou même une distribution).
2. Des **paramètres internes** ( $\theta_i$ ) et un **état** ( $s_i(t)$ ).
3. Un **historique** (ou “mémoire”) de ses interactions passées.

C'est à travers ces entités que s'établissent les **synergies** et que se construit la dynamique de l'apprentissage. En pratique, toute source d'information (une image, un signal audio, un embedding textuel, etc.) peut être encapsulée sous forme d'entité.

**Synergie (ou “Synergy”)** : La **synergie** entre deux (ou plusieurs) entités est la mesure de la **valeur ajoutée** qu'elles obtiennent en coopérant, par rapport à ce qu'elles pourraient réaliser indépendamment (voir 1.2.2). Elle se note souvent  $S(\mathcal{E}_i, \mathcal{E}_j)$  pour les paires, et peut être généralisée à des ensembles  $\{\mathcal{E}_{k_1}, \dots, \mathcal{E}_{k_m}\}$ .

#### Formes de mesure.

- **Informationnelle** : Basée sur l'entropie, l'information mutuelle, ou d'autres métriques de la théorie de l'information.
- **Basée sur la performance** : Différence de score (classification, régression, etc.) quand on associe  $\mathcal{E}_i$  et  $\mathcal{E}_j$ .

- **Hybride** : Combinaison d'un critère d'information et d'un critère de performance.

La synergie est la **“force motrice”** du DSL : elle guide la création, la rupture ou le renforcement des connexions entre entités (voir ci-dessous “pondérations synergiques”).

**Pondérations Synergiques (ou “Synergistic Weights”)** : Les **Pondérations Synergiques** notées  $\omega_{i,j}(t)$ , ce sont les **coefficients** qui caractérisent la relation dynamique entre deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  à l'instant  $t$ .

Souvent modélisée par une équation du type

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

où  $\eta$  est un taux d'apprentissage,  $\tau$  un terme de régularisation, et  $S(\mathcal{E}_i, \mathcal{E}_j)$  la synergie entre les entités.

Les pondérations synergiques constituent la **matrice d'adjacence** d'un *graphe évolutif* :

$$W(t) = [\omega_{i,j}(t)]_{i,j}.$$

Elles déterminent quelles entités sont fortement liées (hautes synergies) et lesquelles le sont moins voire pas du tout (synergie quasi nulle).

**Cluster (ou “Micro-Réseau”)** : Un **cluster** est un **sous-ensemble** d'entités qui présentent entre elles une synergie élevée, formant ainsi une structure cohérente et **auto-organisée**.

Les entités  $\{\mathcal{E}_1, \dots, \mathcal{E}_k\}$  tendent à se regrouper si leurs **pondérations synergiques** mutuelles sont supérieures à un certain seuil  $\theta$ , ou si elles maximisent un critère global (p. ex. somme des synergies internes au cluster).

Les **clusters synergiques** formés au sein d'un **DSL** jouent un **rôle** essentiel à deux niveaux. D'abord, ils **favorisent** la **coopération locale**, en permettant aux entités d'un même cluster d'échanger de manière intensive ; chaque entité contribue ainsi ses données ou compétences spécifiques, renforçant la synergie collective.

Ensuite, ils **facilitent** l'**adaptation** du réseau : ces clusters peuvent en effet fusionner pour gérer de nouveaux contextes si leur compatibilité s'avère élevée, ou se scinder lorsqu'un manque de synergie interne se manifeste.

Grâce à ce double mécanisme — coopération accrue et flexibilité structurelle —, le système demeure résilient et à même d'évoluer face aux changements de tâches ou d'environnements.

**Synergistic Connection Network (SCN)** : Le **SCN** représente l'**infrastructure** du DSL. C'est un **réseau** dont les *nœuds* sont les entités  $\{\mathcal{E}_i\}$  et dont les *arêtes* sont les pondérations  $\{\omega_{i,j}\}$ .

Contrairement à un réseau de neurones statique, le SCN est **dynamique** : au fil du temps (ou au fil des itérations d'apprentissage), de nouvelles connexions apparaissent, d'autres se suppriment ou s'affaiblissent, et des clusters émergent.

L'**objectif** central du SCN consiste à **exploiter** les **synergies** entre entités de manière à *auto-organiser* le flot d'information et, ce faisant, à optimiser la **performance** globale du système, qu'il s'agisse d'une tâche supervisée ou non supervisée. L'idée est de permettre aux liens synergiques les plus pertinents de se renforcer, afin que le réseau dirige spontanément les informations vers les chemins les plus efficaces. Ainsi, l'architecture se réagence en fonction des besoins (ou des données) pour offrir un apprentissage et un traitement des informations plus rapide et plus robuste, sans nécessiter de contrôle externe permanent.

**Auto-Organisation : Auto-organisation** désigne la **capacité** d'un réseau à *se structurer* et *se reconfigurer* de façon autonome, sans intervention ou contrôle direct de l'extérieur (cf. section 1.2.4). Ce phénomène repose sur une **évaluation continue** de la synergie entre l'ensemble (ou une partie) des entités : à chaque itération, les pondérations  $\omega_{i,j}$  sont **mises à jour** selon une règle d'adaptation, et des **clusters** peuvent se **former** ou se **dissoudre** en fonction des tendances observées.

L'**objectif** de ce mécanisme est triple. D'abord, il s'agit d'acquérir une **robustesse** accrue face aux perturbations, car le réseau peut se réorganiser spontanément lorsque des défaillances ou des changements surviennent. Ensuite, cette approche permet de gérer naturellement la **multimodalité** : au lieu de cloisonner les entités (visuelles, auditives, etc.), on les laisse s'associer ou se séparer au gré de leurs synergies. Enfin, l'auto-organisation ouvre la voie à un **apprentissage continu**, dans lequel de nouvelles représentations émergentes se forment au fil du temps, sans imposer la rigidité d'un schéma hiérarchique figé.



**État (ou “State”) d’une Entité** : Chaque entité  $\mathcal{E}_i$  dispose d’un **état interne**  $\mathbf{s}_i(t)$ , souvent représenté par un vecteur de dimension  $d$ , qui synthétise son “histoire” ou son “contexte” à l’instant  $t$ . Cet état évolue selon une **fonction d’actualisation**  $F$ , de la forme

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

inspirée, par exemple, de modèles dynamiques ou de mécanismes de type RNN ou “Hebb étendu”. Le **rôle** de  $\mathbf{s}_i(t)$  est déterminant pour la réactivité de l’entité : une entité ayant déjà établi de fortes coopérations avec une autre  $\mathcal{E}_j$  est généralement plus encline à **se synchroniser** de nouveau avec elle, la mémoire de ses interactions passées renforçant la probabilité d’une synergie future.

**Mécanismes de Fusion et de Dissociation** : Deux (ou plusieurs) entités  $\{\mathcal{E}_i, \mathcal{E}_j, \dots\}$  peuvent **fusionner** s’il s’avère qu’elles sont presque systématiquement dans un même cluster et qu’elles partagent une forte synergie dans la durée. Cette fusion se modélise par la création d’une **nouvelle entité**  $\mathcal{E}_{\text{fusion}}$ , qui combine leurs états, leurs mémoires et leurs représentations.

Lorsqu’une entité  $\mathcal{E}_k$  se trouve dans un cluster peu cohérent (synergie moyenne ou négative), elle peut se **retirer** du cluster ou rompre une fusion antérieure.

Ces mécanismes confèrent au DSL une **plasticité structurale** comparable à celle de certains systèmes biologiques (cerveau, colonies d’insectes, etc.), favorisant l’adaptation face à de nouveaux contextes ou de nouvelles tâches.

**Énergie ou Fonction Globale  $\mathcal{J}$**  : On peut parfois définir une **fonction**  $\mathcal{J}(G)$  — parfois appelée “énergie libre” ou “coût global” — qui regroupe, d’une part, les **synergies** positives entre les entités et, d’autre part, un **terme** de pénalisation destiné à éviter une **surabondance** de connexions. Par exemple :

$$\mathcal{J}(G) = - \sum_{i,j} \omega_{i,j} S(\mathcal{E}_i, \mathcal{E}_j) + \alpha \|\boldsymbol{\omega}\|^2,$$

où  $\boldsymbol{\omega}$  désigne le vecteur de toutes les pondérations synergiques, et  $\alpha$  un coefficient de régulation. **Minimiser**  $\mathcal{J}(G)$  revient alors à **maximiser** la somme de synergies utiles tout en **limitant** la prolifération de liens non pertinents. Il s’agit ainsi d’une démarche **globale** pour piloter l’auto-organisation du réseau, puisqu’elle encourage les connexions réellement productives tout en imposant un frein à celles qui n’apporteraient aucun gain substantiel.

**Apprentissage Continu (ou “Lifelong Learning”)** : Dans le **DSL**, l’apprentissage ne se limite pas à une **phase offline** unique ; le réseau peut, au contraire, **évoluer** continuellement face à un flux de données **online**, en réajustant de façon permanente les pondérations  $\omega_{i,j}(t)$  ainsi que la configuration des clusters.

Cet **apprentissage continu** présente plusieurs **avantages** : d’une part, il offre une **tolérance** accrue aux perturbations (bruit, changements dans la distribution des données, apparition de nouvelles classes ou contextes), ce qui lui permet de s’adapter plus aisément à des environnements non stationnaires. D’autre part, il contribue à la **réduction** du phénomène de “forgotten knowledge”, puisque les clusters formés pour des tâches antérieures peuvent être préservés et ainsi servir de base à des transferts de connaissances ultérieurs.

## Terminologies récurrentes (Synthèse)

Pour faciliter la lecture et l’implémentation, voici un **récapitulatif** des principales terminologies :

1.  $\mathcal{E}_i$  : Entité d’information numéro  $i$ .
2.  $S(\mathcal{E}_i, \mathcal{E}_j)$  : Synergie entre entités  $i$  et  $j$ .
3.  $\omega_{i,j}(t)$  : Pondération synergique (ou lien) entre entités  $i$  et  $j$  à l’instant  $t$ .
4.  $W(t) = [\omega_{i,j}(t)]$  : Matrice (ou graphe) des pondérations synergiques.
5. **Cluster** : Sous-groupe d’entités fortement liées (hautes  $\omega_{i,j}$ ).
6. **SCN** : Synergistic Connection Network, la structure dynamique qui évolue selon les lois d’adaptation.

7. **Auto-organisation** : Processus par lequel la structure  $W(t)$  se réarrange spontanément.
8. **Fonction  $J$**  : Mesure globale de la qualité ou de l'état du réseau (peut inclure la somme des synergies, des pénalités, etc.).

## Conclusion

Ces termes — **entité**, **synergie**, **pondération synergique**, **cluster**, **SCN**, **auto-organisation**, etc. — forment le *lexique de base* du DSL. Chaque concept y est interdépendant : les **entités** interagissent via des **pondérations synergiques** qui façonnent le **SCN**, lequel se **réorganise** en **clusters** au gré d'un mécanisme d'**auto-organisation** orienté par la **synergie** et, éventuellement, par une **fonction globale  $J$** .

Dans la section suivante (1.2.7), nous illustrerons ces principes par des **exemples concrets**, qu'ils proviennent de la nature (inspirations biologiques) ou d'applications pratiques (cas d'études multimodales, émergence de schémas cognitifs, etc.). Ce sera l'occasion de vérifier comment l'utilisation rigoureuse de cette terminologie peut clarifier la **logique** et la **mise en œuvre** du Deep Synergy Learning.

### 1.2.7. Exemples Illustratifs de la Synergie dans la Nature

Les principes de **synergie informationnelle** et d'**auto-organisation** que promeut le Deep Synergy Learning (DSL) trouvent de nombreux échos dans les systèmes naturels. Qu'il s'agisse de colonies d'insectes, de réseaux neuronaux biologiques, d'écosystèmes ou de synchronisations collectives, on observe des processus où l'**ensemble** dépasse la **somme de ses parties**, grâce à des mécanismes coopératifs distribués. Les sous-sections suivantes illustrent comment ces phénomènes naturels inspirent l'approche synergiques du DSL.

#### Colonies d'Insectes et Intelligence Collective

Les colonies de fourmis, d'abeilles ou de termites constituent des exemples emblématiques d'**intelligence collective**, car chaque individu, aux capacités limitées, contribue à la réalisation de tâches pourtant très élaborées : construction de nids sophistiqués, optimisation de la recherche de nourriture, etc. Ces interactions reposent sur des **signaux locaux** (phéromones, contacts antennaires, etc.), sans qu'aucune entité centrale ne dirige l'ensemble. L'émergence d'une organisation globale, comme le traçage de pistes ou la réparation du nid, résulte donc d'une **coopération** distribuée entre entités locales.

Dans le **Deep Synergy Learning (DSL)**, les **entités d'information** jouent un rôle comparable à celui de ces insectes : elles établissent ou rompent des liens en fonction de la **pertinence** (ou synergie) qu'elles y perçoivent. Ces **connexions synergiques** évoluent sans cesse, à l'image des fourmis qui renforcent ou abandonnent certains chemins selon leur utilité.

La formation de "clusters" d'entités synergiques dans le DSL évoque les **micro-sociétés** existant au sein d'une colonie d'insectes, où chaque groupe se spécialise dans une tâche particulière. Cette **auto-organisation** spontanée illustre la force d'un système distribué : sans planification rigide, l'ensemble se coordonne pour atteindre un objectif global.

#### Synergies dans le Cerveau et les Réseaux Neuronaux Biologiques

Dans le **cerveau** humain ou animal, la **plasticité synaptique** illustre la puissance d'un réseau extrêmement **connecté**, dans lequel les synapses s'ajustent en fonction des interactions locales. Lorsque deux neurones s'associent régulièrement pour traiter un même stimulus, leur **synapse** se renforce (potentialisation à long terme) ; ce phénomène rappelle la **mise à jour** des liaisons synergiques dans le DSL, où les liens forts se consolident à mesure que les entités coopèrent efficacement.

Les neurosciences démontrent également la formation d'**assemblées neuronales** associées à un concept ou à un stimulus précis. Ces assemblées se créent ou se dissolvent selon le **contexte** ou la **tâche** du moment. De la même manière, le DSL autorise la **création** et la **dissolution** de **clusters** d'entités d'information, la synergie entre ces entités évoluant dans le temps pour s'adapter aux besoins et aux données.

Enfin, ce parallèle s'étend à la **mémoire** et à l'**apprentissage** : dans le cerveau, les synapses permettent à un neurone de "se souvenir" des connexions consolidées lors d'apprentissages antérieurs. Dans le DSL, chaque entité conserve un **état interne** et un **historique** (section 1.2.1), ce qui lui confère une **mémoire contextuelle** et améliore la **cohérence** de l'apprentissage sur le long terme.

## Écosystèmes et Coopérations Symbiotiques

Les **écosystèmes** offrent de nombreux exemples de **coopération** interspécifique : insectes pollinisant les plantes, lichens nés de la symbiose entre algues et champignons, ou encore la mycorhize qui associe champignons et racines de plantes. Dans chacun de ces cas, les organismes trouvent un **bénéfice mutuel**, qu’il s’agisse d’un accès accru aux ressources, d’une protection renforcée ou d’une capacité d’adaptation élargie. C’est la **notion de “gain commun”** qui se déploie ici, illustrant la **synergie** : la coexistence de deux entités (ou espèces) génère une **valeur ajoutée** que l’on ne retrouverait pas si elles agissaient isolément.

Dans le **Deep Synergy Learning (DSL)**, ce principe s’incarne à travers la **mesure de synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  (voir section 1.2.2) et l’**ajustement** adapté des pondérations  $\omega_{i,j}(t)$ . De la même façon que deux espèces coopérantes se renforcent l’une l’autre, deux entités informationnelles voient leurs liens se consolider lorsqu’elles interagissent efficacement. Les **écosystèmes** diversifiés, riches en symbioses, font preuve d’une **résilience** considérable face aux menaces comme la sécheresse ou la prédation, grâce aux ressources complémentaires que chaque espèce apporte. Un **réseau synergique** (DSL) réunissant des entités variées (visuelles, textuelles, auditives, etc.) gagne, lui aussi, en **robustesse** et en **flexibilité** : en modulant continuellement ses interactions, il peut mieux réagir aux imprévus ou aux évolutions de l’environnement.

## Synchronisation Collective : Bancs de Poissons et Nuées d’Étourneaux

Les bancs de poissons et les vols d’oiseaux, tels que les nuées d’étourneaux, illustrent un phénomène de synchronisation remarquable : de larges groupes se meuvent de façon presque chorégraphiée, sans chef unique. Chaque individu ajuste sa trajectoire en fonction de celle de ses voisins, engendrant ainsi un **effet émergent** de cohésion et d’harmonisation.

Ce type d’organisation se comprend généralement à travers quelques **règles simples** (alignement de la vitesse, distance de sécurité, attraction) qui, une fois agrégées, aboutissent à des comportements collectifs complexes. Dans le cadre du Deep Synergy Learning (DSL), la **synergie** et la **mise à jour** des liens jouent un rôle équivalent : lorsque la coopération entre deux entités se révèle bénéfique, celles-ci se synchronisent, et le **réseau global** s’en trouve optimisé.

Les bancs ou les nuées font également preuve d’une grande **plasticité**, se reconfigurant rapidement face à un prédateur ou un obstacle. De façon parallèle, un **réseau synergique** peut, à tout instant, **adapter** sa structure dès lors que le contexte ou les données évoluent, sans nécessiter de “réentraînement” global et figé.

## Conclusion

Les exemples précédents – colonies d’insectes, cerveau, écosystèmes ou synchronisations collectives – démontrent que la synergie émerge lorsque des entités locales et relativement simples **coopèrent** selon des **règles d’interaction** et d’**adaptation**. Sans supervision centrale et sans plan préconçu, il se forme souvent des **structures** ou des **comportements** remarquablement organisés et robustes, capables de s’ajuster aux contraintes du milieu.

Ces observations, empruntées à la nature, guideront la formulation plus formelle des **algorithmes** et la mise en place de **protocoles d’évaluation** pour le DSL. Dans les chapitres suivants, nous verrons comment concrétiser ces analogies sous la forme de modèles mathématiques, de règles de mise à jour et d’applications pratiques, visant à faire du Deep Synergy Learning un **paradigme opérationnel** pour une **IA forte** (ou du moins plus autonome et plus générale).

### Problème 1 : Caractérisation Formelle des Entités d'Information

On considère la section 1.2.1, où une **entité d'information**  $\mathcal{E}_k$  est définie comme un objet pouvant inclure une représentation  $\mathbf{x}_k$ , un état interne  $\mathbf{s}_k$  et un ensemble de paramètres  $\theta_k$ .

#### 1. Représentation mathématique

Proposez une écriture formelle pour l'entité  $\mathcal{E}_k$ , par exemple :

$$\mathcal{E}_k = (\mathbf{x}_k, \mathbf{s}_k, \theta_k),$$

et montrez comment on peut généraliser  $\mathbf{x}_k$  à un **espace de Hilbert** (au lieu de  $\mathbb{R}^d$ ). Quelles propriétés (norme, produit scalaire) sont requises pour permettre des mesures de similarité ?

#### 2. Évolution de l'état interne

Supposons qu'on introduise une fonction de transition :

$$\mathbf{s}_k(t+1) = F(\mathbf{s}_k(t), \theta_k, \mathbf{x}_k, \dots).$$

Décrivez quelles hypothèses de **continuité** ou de **lipschitzianité** (si on veut prouver une existence et unicité de  $\mathbf{s}_k(t)$ ) sont nécessaires.

#### 3. Distances et similarités entité-entité

Pour deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ , on définit une distance :

$$d(\mathcal{E}_i, \mathcal{E}_j) = \alpha \|\mathbf{x}_i - \mathbf{x}_j\| + \beta \|\mathbf{s}_i - \mathbf{s}_j\| + \gamma D(\theta_i, \theta_j),$$

où  $D$  est une distance paramétrique (p. ex. la somme des distances des paramètres).

- Montrez que si  $\|\cdot\|$  et  $D(\cdot, \cdot)$  sont des distances, alors  $d$  en est aussi une, sous conditions  $(\alpha, \beta, \gamma > 0)$ .
- Discutez de l'éventuelle non-linéarité de  $D(\theta_i, \theta_j)$  (p. ex. si  $\theta$  est lui-même un paramètre structural).

#### • Entités comme agents actifs

On suppose qu'une entité peut "agir" en modifiant  $\mathbf{x}_k, \mathbf{s}_k, \theta_k$ . Proposez un cadre formel où l'"action"  $\mathbf{a}_k(t)$  d'une entité influe sur son état futur. Quels problèmes d'analyse mathématique se posent pour étudier la **cohérence globale** de multiples entités interagissantes ?

### Problème 2 : Mesures de Synergie Informationnelle

La section 1.2.2 introduit la notion de **synergie** entre deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . On peut la définir via l'information mutuelle, la corrélation inversée, ou un gain de performance.

#### 1. Axiomes de la Synergie

Proposez quatre propriétés (analogues aux axiomes de l'information mutuelle) qu'une fonction  $S(\mathcal{E}_i, \mathcal{E}_j)$  devrait vérifier pour être considérée comme une "vraie" synergie (ex. non-négativité, symétrie, etc.). Discutez de la compatibilité avec une définition de type "gain de performance".

#### 2. Lien avec l'entropie

Si  $\mathbf{X}_i$  et  $\mathbf{X}_j$  sont les variables aléatoires associées à  $\mathcal{E}_i, \mathcal{E}_j$ , on définit :

$$S(\mathbf{X}_i, \mathbf{X}_j) = I(\mathbf{X}_i \wedge \mathbf{X}_j) - R(\mathbf{X}_i, \mathbf{X}_j),$$

où  $I$  est l'information mutuelle et  $R$  mesure la "redondance". Justifiez pourquoi on peut avoir  $S > 0$  même si  $I(\mathbf{X}_i, \mathbf{X}_j)$  est faible, dans le cas où la combinaison  $\mathbf{X}_i + \mathbf{X}_j$  apporte un bénéfice supérieur à la somme des bénéfices individuels.

#### 3. Maximisation de la synergie

Envisagez un problème d'**optimisation** où l'on cherche à trouver des transformations  $f_i$  et  $f_j$  (appliquées respectivement aux représentations de  $\mathcal{E}_i$  et  $\mathcal{E}_j$ ) pour **maximiser**  $S(f_i(\mathcal{E}_i), f_j(\mathcal{E}_j))$ .

- Formulez le problème sous forme d'un **argmax**
- Discutez de l'existence potentielle de "solutions triviales" (par ex. tout mapper vers une même constante).
- **Généralisation à la synergie n-aire**  
Exposez une définition possible de  $S(\mathcal{E}_1, \dots, \mathcal{E}_n)$ . Quelles difficultés conceptuelles (et calculatoires) surgissent si on veut éviter la simple somme des synergies binaires, afin de capter les effets strictement "collectifs" ?
- **Équivalence synergie-corrélation non linéaire**  
Montrez un exemple (même purement théorique) où deux entités  $\mathcal{E}_i, \mathcal{E}_j$  ont une corrélation linéaire nulle, mais une synergie positive. Quel type de fonction non linéaire relie  $\mathbf{x}_i$  et  $\mathbf{x}_j$  pour produire ce phénomène ?

### Problème 3 : Interaction, Synergie et Corrélacion

La section 1.2.3 aborde la **différence** entre ces trois notions : interaction, synergie, corrélation.

#### 1. Relation formelle interaction vs. Synergie

Montrez, via un exemple ou un énoncé mathématique, qu'une **forte interaction** (modèle non linéaire où la sortie dépend de  $\mathbf{x}_i \times \mathbf{x}_j$ ) n'implique pas forcément une **synergie positive** au sens "gain commun". On peut utiliser un modèle polynomial pour illustrer cela.

#### 2. Corrélacion et redondance

Dans le cas d'un vecteur  $\mathbf{X} = (X_1, X_2)$  gaussien, rappelez la formule de la corrélation linéaire  $\rho$ . Montrez que si  $\rho$  est proche de 1 (corrélation élevée), alors le gain de performance en combinant  $X_1$  et  $X_2$  (pour prédire une variable  $Y$ ) est faible, relevant d'une **redondance** plutôt que d'une **synergie**.

#### 3. Synergie mesurée par un gain de performance

On définit :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \Phi(\{\mathcal{E}_i, \mathcal{E}_j\}) - [\Phi(\{\mathcal{E}_i\}) + \Phi(\{\mathcal{E}_j\})],$$

où  $\Phi$  est un score de prédiction (classification, etc.). Élaborez les conditions nécessaires pour que cette définition soit non triviale (par ex. si la somme  $\Phi(\{\mathcal{E}_i\}) + \Phi(\{\mathcal{E}_j\})$  n'est pas déjà maximale).

#### 4. Lien avec l'information mutuelle conditionnelle

Proposez une **formulation** reliant l'information mutuelle conditionnelle  $I(\mathbf{X}_i, \mathbf{X}_j | \mathbf{Y})$  à l'idée de synergie, où  $\mathbf{Y}$  serait la variable cible (label). Dans quelles hypothèses peut-on voir cette conditionnelle comme un indicateur de synergie ?

### Problème 4 : Hiérarchie vs. Auto-Organisation

La section 1.2.4 compare l'**approche hiérarchique** (couches fixes) à une **approche auto-organisée** (où les liens se créent/détruisent selon la synergie).

#### 1. Modèle hiérarchique "statique"

Considérez un réseau classique (feed-forward) de  $L$  couches. Définissez la fonction de propagation :

$$\mathbf{h}^{(1)} = f^{(1)}(\mathbf{x}), \quad \mathbf{h}^{(2)} = f^{(2)}(\mathbf{h}^{(1)}), \dots, \mathbf{h}^{(L)} = f^{(L)}(\mathbf{h}^{(L-1)}).$$

Quelle est la dimension de l'espace paramétrique ? Pourquoi dit-on que l'architecture n'est pas modifiée par l'apprentissage (seuls les poids le sont) ?

#### 2. Graphe évolutif

Dans une approche auto-organisée, on modélise le réseau par un graphe  $G(t)$  à  $n$  entités, avec des pondérations  $\omega_{i,j}(t)$ . Établissez des équations type :

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

Montrez comment ce système peut dépasser la simple structure de couche “1 vers 2 vers 3...”.

### 3. Critère de “profondeur”

Dans un réseau hiérarchique, la notion de “profondeur” est définie par le nombre de couches. Comment définir une notion équivalente de “profondeur” ou de “distance” dans un graphe auto-organisé ? (Ex. longueur moyenne des chemins entre entités, diamètre, etc.) Quels problèmes de complexité cela pose-t-il si le graphe évolue en temps réel ?

### 4. Cas limite : le graphe complet

Que se passerait-il si, dans une approche auto-organisée, toutes les pondérations  $\omega_{i,j}(t)$  devenaient fortes ? Discuter d’un point de vue mathématique pourquoi une telle situation est souvent **instable** ou pénalisée (soit par un mécanisme d’énergie libre, soit parce que l’on observerait alors du bruit redondant).

## Problème 5 : Réseaux Synergiques et Exemples Naturels

La section 1.2.7 présente des **exemples** de synergie dans la nature (cerveau, colonies d’insectes, écosystèmes...).

### 1. Equation de type Hebb pour la plasticité synaptique

En neurosciences, la mise à jour “hebbienne” peut être écrite (simplifiée) :

$$w_{ij}(t+1) = w_{ij}(t)\eta[a_i(t)a_j(t)],$$

où  $a_i(t)$  est l’activité du neurone  $i$ . Montrez en quoi c’est une **approximation** d’une règle de synergie  $\omega_{i,j}(t+1) \approx \omega_{i,j}(t) + [S_{i,j}] - \tau\omega_{i,j}(t)$ . Quelles hypothèses simplificatrices implique l’absence d’un terme  $-\tau w_{ij}$  ?

### 2. Modélisation mathématique des colonies d’insectes

Dans la recherche de nourriture, le trajet d’une fourmi peut se modéliser par un **champ de phéromone**  $\phi_{ij}$  sur l’arête  $(i, j)$  d’un graphe représentant le terrain. Établissez la similitude avec le DSL :

$$\phi_{ij} \leftrightarrow \omega_{i,j},$$

loi d’évaporation  $\phi_{ij}(t+1) = (1-\delta)\phi_{ij}(t)$  faisant écho au terme  $-\tau\omega_{i,j}(t)$ . Discuter des contraintes mathématiques pour prouver la convergence vers un chemin optimal.

### • Auto-régulation dans un écosystème

Imaginez  $n$  espèces, chacune ayant une population  $x_i(t)$ . Les interactions peuvent être modélisées par un système de Lotka–Volterra couplé :

$$\frac{dx_i}{dt} = x_i \left( \alpha_i + \sum_{j \neq i} \beta_{i,j} x_j \right),$$

où  $\beta_{i,j}$  peut être positif (synergie) ou négatif (compétition). Montrez en quoi cela illustre la **coévolution** d’entités, et comparez cette dynamique à la formation/dissolution de liens  $\omega_{i,j}$  dans le DSL.

### • Synchronisation de bancs de poissons

Considérez un **modèle de Vicsek** (ou équations similaires) où chaque entité (poisson, oiseau) met à jour sa vitesse en se rapprochant de celle de ses voisins. Comment le phénomène d’“alignement local” peut-il être vu comme un **cas particulier** de synergie,  $S(\mathcal{E}_i, \mathcal{E}_j)$  ? Décrivez la traduction mathématique via un terme  $\omega_{i,j}(\mathbf{v}_j - \mathbf{v}_i)$ .

### • Conclusion sur la transposition aux “réseaux synergiques”

Pour chacun des exemples (cerveau, fourmis, écosystème, synchronisation), reformulez succinctement comment on passe d’une description “naturelle” (neuronale, biologique, etc.) à une description “inspirée DSL” (entités d’information, liens

synergiques, lois de mise à jour). Quels **défis** (analyse de stabilité, attracteurs multiples, etc.) subsistent pour établir des **résultats formels** sur la convergence ou la robustesse des réseaux synergiques ainsi modélisés ?



### 1.3. Importance de l'Auto-Organisation

Dans le cadre du **Deep Synergy Learning (DSL)**, l'un des aspects les plus novateurs réside dans sa capacité d'**auto-organisation**. Alors que les approches d'IA traditionnelles s'appuient souvent sur des architectures hiérarchiques fixes, le DSL encourage un processus dynamique où les **entités d'information** établissent et révisent en continu leurs relations, en fonction de la **synergie** qu'elles détectent entre elles (voir sections précédentes). Cette section (1.3) aborde l'importance de l'auto-organisation sous plusieurs angles, depuis les sources d'inspiration biologiques (1.3.1) jusqu'à l'impact sur la résilience des modèles (1.3.6) et les perspectives globales (1.3.7).

#### 1.3.1. Inspirations Biologiques et Cognitives

L'auto-organisation, au sens où l'entend le DSL, n'est pas un concept nouveau. En réalité, de nombreux **systèmes naturels** (organismes vivants, écosystèmes, cerveau, etc.) démontrent d'extraordinaires facultés d'organisation **spontanée** et **adaptative**. Le **Deep Synergy Learning** cherche à exploiter ces principes, en transposant aux systèmes informatiques des mécanismes qui ont fait leurs preuves dans le vivant.

##### Le cerveau comme paradigme de plasticité



Le cerveau humain compte des centaines de milliards de neurones, reliés par des synapses dont les forces (poids synaptiques) évoluent selon l'activité, un phénomène désigné par "plasticité synaptique". On peut symboliser ce principe via une équation de type Hebbien généralisée, par exemple :

$$w_{ij}(t+1) = w_{ij}(t) + \eta [A_{ij}(t) - \lambda w_{ij}(t)],$$

où  $w_{ij}$  est la "force" de la synapse reliant le neurone  $i$  au neurone  $j$ ,  $\eta$  un taux d'apprentissage,  $A_{ij}(t)$  l'activité conjointe des neurones  $i$  et  $j$  (ou une mesure de leur association), et  $\lambda$  un terme de régulation. L'important est de voir comment la synapse se renforce si l'activité commune est significative, rappelant la **mise à jour** des **pondérations synergiques** dans le DSL (cf. section

1.2.6).

Dans le cerveau, des regroupements (ou *assemblées*) de neurones se forment lorsque leurs synapses se renforcent mutuellement. Le concept de cluster auto-organisé dans le DSL (où les entités se regroupent pour exploiter leur synergie) ressemble à cette idée :

$$C^* = \operatorname{argmax}_{C \subseteq \{\mathcal{E}_1, \dots, \mathcal{E}_N\}} \sum_{(i,j) \in C \times C} \omega_{i,j},$$

c'est-à-dire la recherche d'un sous-ensemble  $C$  maximisant la somme des pondérations entre ses membres, analogiquement à des neurones qui s'assemblent pour traiter un même stimulus.

La cognition résulte du fait que certains assemblages persistent dans le temps et renforcent leur "*autoroute synaptique*" (par exemple, mémorisation d'un stimulus répété). Dans le DSL, l'émergence de **micro-réseaux** stables autour d'entités synergiques reflète ce même phénomène d'**apprentissage distribué**, favorisant la récurrence de schémas pertinents.

##### 1.3.1.2. Colonies d'insectes et intelligence collective



Les colonies de fourmis s'illustrent souvent par leur capacité à trouver des chemins optimaux pour la recherche de nourriture. Ce comportement repose sur des **interactions locales** (dépôt de phéromones, etc.), et non sur un plan global. L'analogie avec le DSL se fait sentir lorsque l'on considère que chaque fourmi incarne une "entité" prenant des décisions locales qui, agrégées, produisent une optimisation globale.

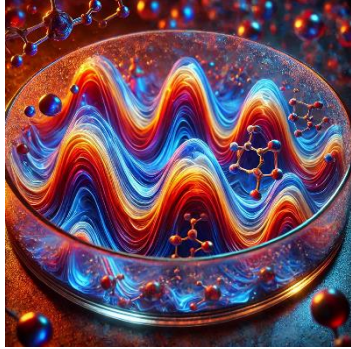
On peut modéliser la concentration de phéromones le long d'un chemin par une fonction  $\phi(t)$  obéissant à un mécanisme de renforcement/dissipation, par exemple :

$$\frac{d\phi}{dt} = \alpha I(\text{fourmis}) - \beta \phi,$$

où  $\alpha$  modélise l'ajout de phéromones (si des fourmis passent) et  $\beta$  la vitesse d'évaporation. Le DSL retrouve un concept similaire quand il gère l'**intensité** d'une liaison synergique via des règles de renforcement ou de décroissance (cf. équations de mise à jour des pondérations, section 1.2.4).

À l'instar de chemins de phéromones qui "naissent" (s'ils sont utiles) et "disparaissent" (s'ils ne sont plus empruntés), les connexions synergiques  $\omega_{ij}(t)$  du DSL se renforcent ou s'étiolent, entraînant la **création** ou la **rupture** de clusters d'entités d'information. Cette dynamique **auto-organisée** ne requiert pas de commande centrale.

### 1.3.1.3. Systèmes dynamiques et attracteurs auto-organisés



De nombreux systèmes auto-organisés (réacteurs chimiques de type Belousov-Zhabotinsky, réactions oscillantes, etc.) peuvent s'exprimer par des **équations différentielles** à plusieurs variables, où les interactions locales créent des **boucles de rétroaction** positives ou négatives. Les motifs qui émergent (ondes chimiques, spirales...) sont des exemples d'**attracteurs** dynamiques.

Dans le DSL, on peut considérer que l'**état global** du réseau est donné par la matrice  $\Omega(t) = [\omega_{ij}(t)]$ . L'évolution de  $\Omega(t)$  suit des règles de type :

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

Cette équation peut se voir comme un **système dynamique** discret, susceptible de converger vers un attracteur (une configuration stable) ou d'entrer dans des régimes oscillatoires, reflétant l'**auto-organisation**.

Dans un cadre mathématique plus poussé, on peut étudier la **stabilité** de l'état  $\Omega^*$  en résolvant :

$$\omega_{i,j}(t+1) - \omega_{i,j}(t) = 0,$$

pour tous  $(i, j)$ . On obtient alors une condition d'équilibre, ce qui, dans un contexte du DSL, correspond à un réseau où chaque liaison a atteint une valeur stable en regard des synergies (sous réserve que l'on n'introduise pas de nouvelles entités ou de nouvelles données). Une **analyse de la Jacobienne** locale pourrait prédire si cet état  $\Omega^*$  est un **point d'attraction**, un **cycle limite** ou un **chaos**.

(Voir Appendix 1)

### 1.3.1.4. Vers un transfert de ces principes au DSL

Les systèmes biologiques et cognitifs décrits ci-dessus mettent en avant quatre qualités fondamentales :

- **Plasticité** (adaptation continue)
- **Robustesse** (résistance au bruit ou à la perturbation)
- **Emergence** (apparition de structures non planifiées)
- **Auto-régulation** (équilibre entre renforcement et inhibition)

Le DSL s'inspire de ces qualités pour proposer un **apprentissage distribué**, dans lequel les entités d'information se comportent comme des agents qui renforcent leurs liens lorsqu'ils détectent un **gain** (cf. notion de synergie) et les relâchent sinon. Matériellement, on transpose ces mécanismes via :

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

et des **clusters** se forment spontanément si le gain est significatif.

Contrairement aux réseaux neuronaux profonds classiques, où seule la valeur des poids évolue dans une architecture figée, l'**auto-organisation** du DSL laisse la structure du réseau se **recomposer** en fonction des synergies détectées. Les entités elles-mêmes peuvent évoluer, en modifiant leur **représentation** ou leur **état** interne, et donc modifier indirectement la dynamique globale du réseau.

### 1.3.1.5. Conclusion partielle : fondements biologiques de l'auto-organisation

Cette section a illustré, par des références à la **plasticité synaptique**, à l'**intelligence collective** chez les insectes et à divers **systèmes dynamiques**, l'importance des mécanismes d'auto-organisation en biologie et en cognition. Le **Deep Synergy Learning** cherche à exploiter ces mêmes principes pour créer des modèles d'intelligence artificielle moins dépendants d'une structure hiérarchique imposée,

plus adaptatifs, et potentiellement plus proches d'une **IA générale**. Les sections suivantes (1.3.2 à 1.3.7) mettront en évidence des concepts plus généraux d'**émergence**, de **feedback**, et détailleront comment l'auto-organisation du DSL se compare aux méthodes d'apprentissage classiques, avant de conclure sur la pertinence de cette approche pour des architectures globales et multimodales.

### 1.3.2. Concepts Clés : Émergence, Auto-Régulation, Feedback

La dynamique **auto-organisée** que promeut le Deep Synergy Learning (DSL) repose sur plusieurs notions fondamentales, à la fois conceptuelles et mathématiques, qui permettent de comprendre comment un système d'entités peut s'ajuster et se coordonner sans planification centrale. Trois d'entre elles sont particulièrement essentielles : l'**émergence**, l'**auto-régulation** et le **feedback** (rétroaction). Les sous-sections ci-après détaillent la signification de chacun de ces concepts et leur importance dans le cadre du DSL.

#### Émergence

Le terme **émergence** décrit la situation où un **phénomène global** ou une **structure** à l'échelle macroscopique naît de l'interaction de **composants** à l'échelle microscopique, sans qu'il n'y ait de plan explicite ni de chef d'orchestre. Par exemple, la forme d'une nuée d'oiseaux ou l'organisation complexe d'une ruche peuvent être vus comme des propriétés émergentes d'un grand nombre d'interactions locales.

Dans le Deep Synergy Learning, l'**émergence** se manifeste par la formation de **clusters** (ou micro-réseaux) d'entités d'information, que l'on peut qualifier de "sous-systèmes cohérents" au sein du grand réseau synergique. Ces clusters ne sont pas prédéfinis : ils **apparaissent** (et parfois **disparaissent**) en fonction de la synergie détectée entre entités.

Un cluster  $\mathcal{C} \subset \{\mathcal{E}_1, \dots, \mathcal{E}_n\}$  peut s'illustrer par une somme de pondérations internes  $\sum_{(i,j) \in \mathcal{C} \times \mathcal{C}} \omega_{i,j}$  qui se trouve **maximalisée** à un certain moment  $t$ .

De plus, ces clusters émergents peuvent générer de nouvelles **représentations** ou entités "conjointes", lesquelles elles-mêmes nourrissent l'**évolution** du réseau.

Sur le plan formel, on peut lier l'émergence au fait que la **dynamique** des pondérations synergiques  $\omega_{i,j}(t)$  conduit le système vers des **attracteurs** (états stables ou semi-stables). L'apparition d'un cluster correspond alors à un **bassin d'attraction** particulier, où les liens internes se renforcent suffisamment pour maintenir l'entité groupée, un phénomène typique des **systèmes dynamiques complexes**.

#### Auto-Régulation

Dans un système auto-organisé, chaque composant (ou chaque lien) s'**ajuste** en fonction de **règles locales** : si la coopération est jugée bénéfique, le lien augmente ; si elle est néfaste ou inutile, le lien diminue. Ce **mécanisme d'ajustement** relève de l'**auto-régulation**, car il ne dépend pas d'une autorité externe qui dirait "ces entités doivent se connecter, celles-ci doivent se séparer".

Comme décrit en section 1.2.4, une règle fréquemment utilisée dans le DSL pour modéliser l'évolution des pondérations synergiques  $\omega_{i,j}(t)$  est :

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

- $\eta$  représente un taux d'apprentissage (contrôle la rapidité de l'ajustement).
- $S(\mathcal{E}_i, \mathcal{E}_j)$  est la **synergie** calculée entre les entités  $i$  et  $j$ .
- $\tau$  agit comme un terme de décroissance ou de "frottement", évitant que les pondérations ne croissent indéfiniment.

Le paramètre  $\tau$  et le choix de la fonction  $S(\cdot)$  permettent d'**éviter les dérives** (ex. explosion des connexions) et de stabiliser le système dans un régime d'**énergie** ou de **coût** relativement bas (voir fonction  $\mathcal{J}$  discutée en 1.2.6). Ce mécanisme d'auto-régulation fait qu'on n'a pas besoin de redéfinir manuellement la topologie du réseau : elle s'ajuste d'elle-même aux contraintes et aux données entrantes.

#### Feedback (Rétroaction)

Le **feedback** désigne le processus par lequel la **sortie** d'un système ou l'**état** d'une partie du système **retourne** sur lui-même, influençant à son tour la configuration ou le comportement du système. Dans le DSL, le feedback se matérialise surtout à travers l'**apprentissage** des synergies : la façon dont deux entités ont coopéré à l'instant  $t$  affecte la pondération  $\omega_{i,j}(t+1)$ , qui influencera à son tour leur coopération future.

Le **feedback positif** est un mécanisme par lequel une forte synergie entre deux entités, notée  $S(\mathcal{E}_i, \mathcal{E}_j)$ , conduit à un renforcement mutuel de leur connexion. En d'autres termes, si la synergie initiale entre deux entités est élevée, leur interaction devient progressivement plus forte, renforçant encore davantage leur synergie. Ce processus peut continuer jusqu'à atteindre un certain plafond, au-delà duquel la connexion ne peut plus croître, stabilisant ainsi leur relation synergique.

À l'inverse, le **feedback négatif** intervient lorsque la synergie entre deux entités est faible ou nulle, voire négative. Dans ce cas, la pondération de leur lien décroît progressivement, réduisant la probabilité qu'elles interagissent à l'avenir. Cependant, ce mécanisme n'est pas définitif : un changement de contexte ou d'état pourrait, à terme, améliorer leur compatibilité et rétablir leur capacité à collaborer ou à former une connexion.

Outre les liens binaires (entre paires d'entités), il peut exister des **boucles de rétroaction plus complexes** à l'échelle de plusieurs clusters. Par exemple, un cluster  $\mathcal{C}_1$  peut influencer la formation ou la dissolution d'un autre cluster  $\mathcal{C}_2$  en modifiant le contexte, les ressources partagées, etc. Sur le plan mathématique, cela équivaut à considérer des **fonctions de synergie n-aire**, où les "coûts" ou "gains" de chaque liaison dépendent aussi des états des autres liens et entités.

$$S(\mathcal{E}_i, \mathcal{E}_j \mid \{\mathcal{E}_k\}_{k \neq i,j}), \quad \omega_{i,j}(t+1) = f(\omega_{i,j}(t), \{S(\mathcal{E}_i, \mathcal{E}_j \mid \dots)\}, \dots).$$

## Synthèse

Les trois concepts d'**émergence**, d'**auto-régulation** et de **feedback** interagissent étroitement dans le Deep Synergy Learning :

- L'**émergence** se produit parce que les liens (pondérations synergiques) se développent localement sans qu'aucun schéma global ne soit imposé.
- L'**auto-régulation** assure que chaque lien est entretenu ou affaibli en fonction de la **pertinence** mesurée (synergie), maintenant ainsi une forme de **plasticité** dans l'ensemble du réseau.
- Le **feedback** perpétuel, tant au niveau des paires d'entités que des ensembles plus larges, crée un **cycle** où l'état du réseau à l'instant  $t$  influence les coopérations à l'instant  $t+1$ , et ainsi de suite.

Ce mode de fonctionnement "bouclé" et distribué est très différent de la logique hiérarchique classique (voir 1.2.4). Il rappelle davantage les **systèmes biologiques** et autres **systèmes complexes**, où la stabilité et les structures apparaissent comme le fruit d'interactions locales répétées, produisant parfois des schémas surprenants d'ordre (ou de désordre) global.

Comprendre les notions d'**émergence**, d'**auto-régulation** et de **feedback** est crucial pour saisir pourquoi le **DSL** peut offrir des capacités d'adaptation et de créativité supérieures à celles des architectures figées. Loin d'être un simple attribut de "design" théorique, ces concepts constituent la **dynamique interne** qui permet au réseau synergique de se **reconfigurer** en permanence, de **fusionner** ou de **séparer** des entités en clusters, et d'**apprendre** sans devoir figer sa topologie.

Dans la section suivante (1.3.3), nous mettrons en perspective ces mécanismes d'auto-organisation avec les **méthodes d'apprentissage classiques**, afin d'explorer les gains potentiels (et les défis) qu'apporte l'introduction de ces idées dans le champ de l'intelligence artificielle.

### 1.3.3. Comparaison avec les Méthodes d'Apprentissage Classiques

Les principes d'**auto-organisation** et de **synergie** du Deep Synergy Learning (DSL) contrastent fortement avec les schémas employés par les méthodes d'apprentissage les plus courantes, qu'il s'agisse d'algorithmes supervisés, non supervisés ou même de certaines approches en renforcement. Dans la présente section, nous mettons en lumière les principales différences entre ces paradigmes "classiques" (tels que le **Deep Learning** hiérarchique, la **clustering** statique ou la **rétropropagation de gradient** end-to-end) et le **DSL**, qui s'appuie avant tout sur une **construction dynamique** du réseau via la synergie informationnelle.

Dans un **cadre supervisé** classique, on dispose d'un ensemble de données  $\{(\mathbf{x}_i, y_i)\}_{i=1, \dots, N}$ , où  $\mathbf{x}_i$  constitue l'entrée (vecteur de caractéristiques, image, etc.) et  $y_i$  le label associé (catégorie, valeur numérique, etc.). Un **modèle** (par exemple, un réseau de neurones) est alors **entraîné** à minimiser une **fonction de coût** :

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^N L(f_{\theta}(\mathbf{x}_i), y_i),$$

où  $f_{\theta}$  désigne le modèle paramétré par  $\theta$  (poids et biais), et  $L(\cdot)$  une fonction de perte (entropie croisée, MSE, etc.).

La plupart du temps, l'**architecture** du réseau est **figée** : on choisit un nombre de couches, une topologie (CNN, MLP, etc.) et on la conserve durant tout l'entraînement. La **rétropropagation** permet d'ajuster les poids internes  $\theta$ , sans pour autant réorganiser la **structure** (aucune suppression ni création de neurones, aucune reconfiguration des connexions).

Dans un réseau **DSL**, on ne se limite pas à la mise à jour des paramètres : on autorise la **topologie** à évoluer (création, renforcement, fusion, ou rupture de liens) en se fondant sur la **mesure de synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$ . Dans un modèle supervisé classique, la structure reste **inchangée** tout au long de l'apprentissage.

Dans l'approche **supervisée** traditionnelle, l'effort se concentre sur la **réduction de l'erreur** par rapport aux labels  $y$ . À l'inverse, le DSL porte son attention sur la **relation** entre les différentes données elles-mêmes : la **formation** de clusters ou l'**auto-organisation** peut survenir même sans labels, dès que le réseau détecte des synergies fructueuses.

Un **modèle supervisé** doit souvent être **réentraîné** (ou au moins fine-tuné) dès lors qu'apparaissent de nouveaux types de données ou qu'une **distribution** se modifie. Dans le DSL, la structure s'**auto-réorganise** en intégrant ou en écartant certaines entités, en fonction de la synergie relevée, ce qui apporte davantage de **souplesse** et de **plasticité** face aux environnements changeants.

Dans l'apprentissage **non supervisé**, on cherche généralement à regrouper des points  $\mathbf{x}_i$  en clusters. Des méthodes comme **k-means** démarrent avec un nombre de clusters ( $k$ ) et une mesure de distance (par exemple euclidienne ou cosinus), puis procèdent à une série d'itérations visant à minimiser

$$\sum_{i=1}^N \min_{1 \leq c \leq k} \|\mathbf{x}_i - \boldsymbol{\mu}_c\|^2,$$

où  $\boldsymbol{\mu}_c$  sont les centres de clusters. Dans **k-means**, le paramètre  $k$  doit être défini **a priori**, et même si d'autres algorithmes (tels que DBSCAN ou le clustering agglomératif) peuvent estimer le nombre de groupes, ils reposent néanmoins sur une **logique statique**, c'est-à-dire qu'on exécute l'algorithme une fois et on obtient une unique partition.

De plus, dans ces approches classiques, les points — ou vecteurs — ne bénéficient pas d'une **dynamique** interne : ils ne modifient ni leur représentation ni les liens qui pourraient évoluer au fil du temps en fonction d'une collaboration éventuelle. Le DSL se distingue précisément sur ce point : les **clusters** peuvent y **apparaître** ou se **dissoudre** de manière **progressive** selon les pondérations synergiques  $\omega_{i,j}(t)$ , sans qu'un paramètre  $k$  fixe ne borne le regroupement. La structure du réseau se **façonne** ainsi en continu, grâce à la dynamique d'apprentissage.

En outre, chaque entité  $\mathcal{E}_i$  du réseau DSL peut être **active**, c'est-à-dire posséder un **état interne** et évoluer au cours du temps. Dans un clustering **statique**, les points demeurent figés dans leurs coordonnées, sans aucun mécanisme de feedback local ou de synergie émergente. Enfin, lorsqu'apparaissent de nouvelles données ou qu'un changement de distribution survient, le DSL est capable de **réévaluer** les synergies et de **reconfigurer** automatiquement les clusters, tandis qu'un clustering non supervisé traditionnel exigerait la relance complète de l'algorithme, ignorant dans la foulée toute information antérieure.

Dans le **Reinforcement Learning**, un **agent** interagit avec un environnement, perçoit des **récompenses** (positives ou négatives) selon ses actions, et cherche à **maximiser un cumul de récompenses**. Les algorithmes de RL (Q-learning, SARSA, méthodes basées sur une politique) visent souvent à estimer la fonction de valeur  $Q(\mathbf{s}, \mathbf{a})$  ou la politique optimale  $\pi(\mathbf{a}|\mathbf{s})$ . De nombreux scénarios de RL (par ex. Deep Q-Network) s'appuient sur un **réseau profond** paramétré, dont les poids sont ajustés via un **objectif** portant sur la récompense cumulée.

Dans la plupart des implémentations RL traditionnelles, la **structure** du réseau reste fixée, on ne modifie pas, au cours de l'apprentissage, le nombre de couches, ni la forme du modèle.

Alors que le RL privilégie l'action menant à la **maximisation** de la récompense, le DSL promeut la **coopération** (ou la connexion) qui maximise la **synergie**. Il est possible d'envisager un "DSL-RL" où la synergie devient une composante de la **fonction de récompense** : plus la **collaboration** entre entités est jugée fructueuse, plus les liens correspondants se consolident.

Dans le RL classique, on n'ajoute ni ne supprime spontanément des neurones ou des liaisons durant l'apprentissage. À l'inverse, le DSL autorise une **recomposition** de l'architecture si cela contribue à améliorer la performance (ou la cohérence). Les pondérations peuvent émerger, se renforcer, ou disparaître au fil du temps.

On peut voir les entités du DSL comme autant d'"agents partiels" qui s'accordent via la **synergie**, au lieu de recourir à un **agent unique** muni d'un policy network fixe. Cette perspective diffère sensiblement du RL centralisé : l'organisation naît de l'**interaction** locale entre entités, plutôt que de se conformer à une unique stratégie imposée à l'ensemble du réseau.

## Synthèse

Les approches classiques, qu'elles relèvent du **supervisé**, du **non supervisé** ou du **renforcement**, s'appuient généralement sur une **architecture** (ou un **ensemble d'hypothèses**) prédéfinie : il peut s'agir d'un réseau feed-forward, d'un certain nombre de clusters à trouver ou d'une paramétrisation donnée. À l'inverse, le **DSL** ne fixe pas la structure à l'avance ; il permet à celle-ci de **se former** et de **se modifier** de manière autonome, simplement en s'appuyant sur les mesures de **synergie** et sur les **règles** d'auto-organisation.

Dans les méthodes classiques, on privilégie soit le **rapprochement** entre données et labels (apprentissage supervisé), soit la **répartition** de points dans des groupes (apprentissage non supervisé). Le DSL, au contraire, met l'accent sur la **relation** entre les entités, qu'il s'agisse de leur **coexistence** ou de leur **synergie**, et c'est ce mécanisme relationnel qui détermine l'organisation d'ensemble.

Ce renversement de perspective — considérer la donnée non pas comme un **input passif**, mais comme un **acteur** recherchant activement des **connexions** — constitue la marque distinctive du DSL. Les algorithmes traditionnels sont, de surcroît, souvent conçus pour un entraînement **en batch** : on recueille les données, on entraîne le modèle, puis on fige la configuration. Des versions en ligne existent, mais elles n'ont pas pour vocation de retoucher la **structure** du modèle ; elles n'agissent que sur les poids.

Le DSL, quant à lui, se prête naturellement à un **flux continu** de données : à mesure que de nouvelles informations apparaissent, le réseau peut se **reconstruire** pour répondre aux conditions changeantes ou accueillir de **nouvelles entités**, poursuivant ainsi une **adaptation** permanente plutôt que de s'en tenir à un unique schéma déterminé à l'avance.

## Conclusion

La comparaison avec les méthodes classiques souligne le caractère **révolutionnaire** de l'auto-organisation prônée par le DSL : la capacité de faire **“évoluer la carte du réseau”** plutôt que de se restreindre à des architectures fixes, la prise en compte d'une **synergie** mesurée entre entités, la facilité d'**apprentissage continu** et la résistance aux changements de distribution.

Ces différences ne signifient pas que le DSL vienne nécessairement **remplacer** l'ensemble des paradigmes existants. On peut au contraire envisager des **approches hybrides** qui combinent la puissance des algorithmes classiques (supervisés ou RL) avec la **plasticité** et la **coévolution** propres au DSL. Les sections qui suivront (1.3.4 à 1.3.7) approfondiront l'idée d'une **auto-organisation multimodale**, ainsi que l'**impact** concret de cette approche sur la robustesse et la résilience des modèles, pour finalement ouvrir la voie à une IA plus générale et plus souple.

### 1.3.4. Le Rôle des Flux d'Information Multimodaux

Parmi les avantages majeurs de l'**auto-organisation** telle que proposée dans le **Deep Synergy Learning (DSL)** figure la capacité à gérer efficacement des **flux d'information** variés (image, texte, audio, données sensorielles, etc.). En effet, l'idée de **synergie informationnelle** (développée aux sections 1.2.2 et 1.2.3) s'applique de manière particulièrement féconde dès lors que plusieurs modalités sont en jeu, car chaque flux peut **renforcer** ou **compléter** les autres. Cette sous-section (1.3.4) illustre la manière dont l'auto-organisation des entités d'information s'adapte à la **multi-modalité**, et pourquoi cela peut conférer au DSL une **puissance d'intégration** hors de portée des architectures classiques, souvent cloisonnées.

Dans un système multimodal, on distingue plusieurs sources de données :

- **Modalité visuelle** : images, vidéos (entité  $\mathcal{E}_{\text{visuelle}}$ )
- **Modalité auditive** : signaux sonores, musique, parole (entité  $\mathcal{E}_{\text{auditive}}$ )
- **Modalité textuelle** : séquences de mots, documents, balises sémantiques (entité  $\mathcal{E}_{\text{textuelle}}$ )
- **Modalités sensorielles complémentaires** : capteurs physiques (température, distance, pression), signaux biométriques, etc.

Dans le **Deep Synergy Learning**, chaque flux peut être représenté par une ou plusieurs **entités d'information** (voir section 1.2.1). Ainsi, un flux visuel pourra donner naissance à plusieurs entités – par exemple, une entité pour la “carte de caractéristiques” (feature map) d'une image, une autre pour son histogramme de couleurs, etc. Ces entités ne sont pas cloisonnées dans une couche unique : elles peuvent potentiellement interagir avec toutes les autres entités (visuelles, textuelles, auditives...), en fonction de leur **synergie**.

La **synergie** est particulièrement critique en multi-modalité :

$$S(\mathcal{E}_{\text{visuelle}}, \mathcal{E}_{\text{auditive}}) > 0 \Rightarrow \text{l'information visuelle et l'information auditive se complètent efficacement.}$$

Par exemple, dans une scène vidéo où l'on entend quelqu'un parler, la composante auditive (voix) et la composante visuelle (mouvements de lèvres, expressions faciales) coopèrent pour renforcer la compréhension globale.

Lorsqu'une modalité est dégradée (par exemple, l'audio est parasité par du bruit de fond), une autre modalité (l'image) peut prendre le relais, augmentant la **tolérance aux perturbations**. Dans un **réseau synergique**, cela se traduit par une réduction dynamique du poids  $\omega_{\text{audio,visuel}}(t)$  si l'audio devient momentanément peu fiable, et par le renforcement d'autres liens synergiques.

Les entités associées à différentes modalités peuvent créer des **clusters** communs si leur synergie mutuelle est élevée. Mathématiquement, on peut repérer un cluster  $\mathcal{C}_m$  formé d'entités  $\{\mathcal{E}_{\text{visuelle}}^a, \mathcal{E}_{\text{auditive}}^b, \mathcal{E}_{\text{textuelle}}^c\}$  tel que



$$\sum_{(i,j) \in \mathcal{C}_m \times \mathcal{C}_m} \omega_{i,j} > \theta_{\text{seuil}},$$

où  $\theta_{\text{seuil}}$  est un paramètre symbolisant la cohésion requise pour se constituer en cluster. Ce regroupement favorise l'**intégration sémantique** de différents types de signaux.

Dans un réseau hiérarchique classique, la fusion multimodale est généralement réalisée dans une “couche” dédiée (p. ex. concaténation de features). Dans le DSL, la fusion naît **spontanément** :

$$\omega_{\text{visuelle, textuelle}}(t+1) = \omega_{\text{visuelle, textuelle}}(t) \eta [S(\mathcal{E}_{\text{visuelle}}, \mathcal{E}_{\text{textuelle}}) - \tau \omega_{\text{visuelle, textuelle}}(t)].$$

Plus la coopération entre les deux modalités est jugée bénéfique (ex. augmentation de la performance ou de l'information mutuelle), plus ce lien se consolide.

Le fait qu'un cluster multimodal soit **évolutif** est crucial : en fonction du **contexte**, la modalité audio peut prendre plus ou moins d'importance (ex. : un concert de musique vs. un environnement silencieux). Les liens synergiques inter-modaux varient alors dans le temps, faisant émerger ou disparaître des sous-ensembles cohérents.

On peut également enrichir la mesure de synergie  $S$  par un **contexte**  $\mathbf{c}(t)$ , de sorte qu'il y ait des synergies conditionnelles :

$$S(\mathcal{E}_i, \mathcal{E}_j \mid \mathbf{c}(t)).$$

Dans une scène nocturne, par exemple, la vision est moins fiable, tandis que l'audio reste pertinent. Le contexte “nuit” pourrait diminuer certaines liaisons visuelles et renforcer des liaisons audio ou thermiques, mettant l'accent sur la modalité la plus fiable.

## Exemples

Imaginez un système de surveillance qui reçoit en continu des flux vidéo, audio, et des capteurs de mouvement. Dans une approche classique, il faudrait concevoir à l'avance un schéma de fusion ; dans le DSL, les **entités** correspondant à chaque flux s'**auto-organisent** pour détecter des patrons communs (ex. détection d'un intrus combinant une silhouette anormale et des sons inhabituels).

Le **renforcement** des liens entre un flux “caméra infrarouge” et un flux “microphones” peut se produire si ces deux modalités se complètent pour la détection nocturne.

Dans le cadre d'interfaces multimodales (voix, gestes, expressions faciales), chaque modalité peut être captée par une entité DSL. Si l'utilisateur parle tout en faisant des gestes, le cluster “voix + gestes” obtient un fort score de synergie, permettant de mieux interpréter la signification globale (commande ou intention).

L'auto-organisation permet au système de **s'adapter** aux préférences de l'utilisateur : si celui-ci a une diction peu claire mais des gestes très expressifs, les pondérations vers la modalité gestuelle se renforceront spontanément.

Le DSL peut traiter des couples “image + légende textuelle” : une entité “extrait de texte” peut découvrir qu'elle augmente sa synergie avec une entité “détecteur d'objets” dans l'image, notamment si certains mots clés coïncident avec des formes reconnues.

Les entités “texte” et “image” peuvent alors former un **cluster** spécifique de type “concept visuel + label textuel”, facilitant les tâches d'annotation automatique ou de recherche d'images par mots-clés (et vice versa).

## Forces et défis de l'approche synergique multimodale

L'**absence** de couche de fusion prédéfinie procure une grande **flexibilité** : au lieu d'imposer un schéma de combinaison entre les flux (visuel, audio, texte, etc.), le réseau (DSL) **décide** de lui-même quelles modalités s'avèrent les plus pertinentes à associer. Cette adaptabilité favorise également la **robustesse** : si un flux devient bruité ou se perd (par exemple, un microphone défectueux), le système peut se reconfigurer et compter davantage sur d'autres canaux. Enfin, l'**auto-organisation** ouvre la voie à la **découverte** de combinaisons inédites : deux modalités supposées “peu corrélées” peuvent révéler une forte synergie dans un certain contexte, ce qu'un pipeline classique, plus rigide, aurait pu manquer.

En contrepartie, la **complexité computationnelle** peut monter rapidement à mesure que le nombre de modalités et d'entités augmente : évaluer le **gain de synergie** entre multiples canaux peut devenir coûteux, nécessitant l'emploi d'**heuristiques** ou d'**approximations**. De plus, l'enchaînement rapide de contextes (passer d'une scène de rue à une scène d'intérieur, par exemple) peut induire une **volatilité** élevée des connexions synergiques ; il faut alors recourir à des **coefficients de régulation** ( $\tau$ ) ou à des dispositifs de mémorisation pour **stabiliser** la configuration. Enfin, l'**interopérabilité** entre flux implique de disposer de **représentations** comparables ou d'un espace commun adapté (par exemple, une même métrique pour estimer la similarité entre un vecteur d'image et un vecteur de texte).



## Conclusion

Le **Deep Synergy Learning** apparaît particulièrement adapté à l'intégration de **flux multimodaux**. Là où les méthodes classiques imposent généralement une architecture de fusion fixe (par exemple, concaténer l'output d'un CNN d'image et d'un RNN de texte), le DSL laisse les **entités** (issues de différentes modalités) **explorer** leurs synergies possibles. Les liens synergiques se **crystallisent** ou se **dissolvent** en fonction de leur valeur ajoutée, permettant une **adaptation dynamique** aux contextes et aux sources de bruit.

Cette approche auto-organisée favorise l'**émergence** (voir section 1.3.2) de clusters multimodaux plus ou moins stables, qui peuvent évoluer au fil du temps. Sur le plan applicatif, il en résulte une **robustesse** et une **flexibilité** uniques, aussi bien pour la reconnaissance d'événements complexes, la détection d'anomalies, ou encore la mise en place d'interactions homme-machine riches. Les sections suivantes (1.3.5, 1.3.6, 1.3.7) examineront l'**évolution dynamique** de ces réseaux, l'**impact** sur la **résilience** des modèles, et la perspective d'une **approche plus globale** de l'apprentissage.

### 1.3.5. Évolution Dynamique et Adaptation en Continu

Le **Deep Synergy Learning (DSL)** se distingue des approches classiques de l'IA par sa capacité à laisser les entités d'information et leurs connexions **évoluer** en permanence, au gré de la synergie détectée entre les flux de données. Cette caractéristique d'**évolution dynamique** et d'**adaptation en continu** s'inspire directement de divers phénomènes naturels (cerveau, écosystèmes, colonies d'insectes...) et confère au DSL une **plasticité** inhabituelle dans le domaine de l'apprentissage automatique. Dans cette section, nous analysons pourquoi cette évolution constante est essentielle, quels sont ses principes fondamentaux et comment elle se formalise mathématiquement.

Dans le **DSL**, l'apprentissage ne se cantonne pas à une phase d'entraînement figée ; il se présente comme un **processus d'auto-organisation** continu, actif tant que le réseau demeure en service. Chaque entité peut ainsi **affiner** ou **redéfinir** ses connexions et son rôle lorsqu'affluent de nouvelles données, ou même de nouvelles entités.

Contrairement à un réseau neuronal classique, la **structure** du réseau n'est pas fixée : les pondérations synergiques  $\omega_{i,j}(t)$  peuvent **croître** ou **décroître**, au point de passer sous un **seuil** de rupture  $\omega_{\min}$  qui entraîne la disparition effective de la liaison entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . Ce mécanisme libère des ressources et prévient l'encombrement du réseau. À l'inverse, si une synergie autrefois négligeable devient significative, une **nouvelle liaison** peut se former pour consolider la coopération entre entités (voir la section 1.2.6 concernant les notations).

L'environnement ou la distribution des données peut également fluctuer : augmentation du bruit, apparition de nouvelles classes, transformation de contexte, etc. Dans un schéma statique, il faudrait réentraîner ou procéder à un "fine-tuning" coûteux. Dans le DSL, au contraire, l'**adaptation** est **naturellement** assurée par la mise à jour permanente des **liens synergiques** :

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

Si les informations nouvellement introduites modifient la synergie entre entités, la **topologie** du réseau se **recompose** spontanément, sans nécessité de réapprentissage global.

On peut représenter le réseau synergique à l'instant  $t$  par le graphe  $G(t)$ , constitué de nœuds  $\{\mathcal{E}_1, \dots, \mathcal{E}_n\}$  et d'arêtes pondérées  $\omega_{i,j}(t)$ . L'évolution dans le temps se décrit par :

$$G(t+1) = \mathcal{F}(G(t), \mathcal{D}(t), \Theta),$$

où  $\mathcal{D}(t)$  désigne les **données reçues** à l'instant  $t$  (ou sur la fenêtre  $[t, t + \Delta]$ ), et  $\Theta$  englobe les paramètres de mise à jour (taux  $\eta$ , régularisation  $\tau$ , etc.).

Chaque entité  $\mathcal{E}_i$  peut maintenir un **état**  $\mathbf{s}_i(t)$  permettant d'intégrer une forme de **mémoire** (cf. 1.2.1). Cet état influe sur la synergie :

$$S(\mathcal{E}_i, \mathcal{E}_j) = f(\mathbf{s}_i(t), \mathbf{s}_j(t), \mathcal{D}(t), \dots).$$

On a donc un **système dynamique couplé**, dans lequel la mise à jour des connexions  $\omega_{i,j}(t)$  et celle des états  $\mathbf{s}_i(t)$  interagissent :

$$\begin{cases} \omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S(\mathcal{E}_i, \mathcal{E}_j) - \tau \omega_{i,j}(t)], \\ \mathbf{s}_i(t+1) = \mathbf{s}_i(t) + \gamma g(\mathbf{s}_i(t), \{\omega_{i,k}(t)\}, \dots). \end{cases}$$

On obtient alors un **couplage** entités-réseau qui peut donner naissance à des attracteurs, des cycles limites ou même des phénomènes de bifurcation.

Au-delà de la simple création/suppression de liens, il est possible que deux entités  $\mathcal{E}_a$  et  $\mathcal{E}_b$  **fusionnent** si leur collaboration est très constante, formant une **nouvelle entité**  $\mathcal{E}_{ab}$ . De même, une entité fortement hétérogène peut se **scinder** en plusieurs entités spécialisées si cela améliore la synergie globale. Ces mécanismes se décrivent par des règles de type :

$$\text{Fusion}(\mathcal{E}_a, \mathcal{E}_b) \Leftrightarrow \sum_{t'=t_0}^{t_1} \omega_{a,b}(t') > \delta_{\text{fusion}},$$

avec  $\delta_{\text{fusion}}$  un seuil de stabilité sur une durée  $[t_0, t_1]$ . De tels phénomènes rappellent les “assemblées neuronales” durables ou la “spéciation” dans un contexte évolutionniste.

De nombreux problèmes réels (prévision financière, capteurs industriels, suivi de l’activité cérébrale, etc.) impliquent des distributions de données qui **changent** dans le temps. Le DSL, grâce à son **évolution dynamique**, s’ajuste progressivement plutôt que de s’en tenir à un modèle figé.

Dans un réseau traditionnel, apprendre une nouvelle tâche (ou un nouveau domaine) peut faire oublier les acquis précédents, faute de mécanisme de préservation structurelle. Dans le DSL, rien n’oblige à redémarrer l’optimisation : les clusters utiles peuvent **persister**, tandis que de nouvelles entités se lient et modulent la synergie pour assimiler un contexte différent.

Au fur et à mesure de l’évolution, si certaines liaisons n’apportent plus de gain, elles s’effacent ou diminuent. Ce phénomène **régule** la complexité du réseau en évitant une explosion exponentielle du nombre de liens. On peut donc voir le DSL comme un réseau à **topologie parcimonieuse** autorégulée.

## Exemples

Un **système DSL** qui reçoit en continu des mesures (température, pression, vibrations) peut servir à détecter des anomalies ou à anticiper des pannes. Lorsque les conditions d’usage se modifient progressivement (usure, évolution de la chaîne de production), le réseau **réorganise** ses liens synergiques : certains capteurs découvrent une complémentarité avec d’autres, pendant que certains liens historiques deviennent moins utiles et s’affaiblissent. Cette **auto-organisation** permet de maintenir une alerte fiable, sans qu’il soit nécessaire de procéder à un nouvel entraînement global à intervalles réguliers.

Un **chatbot multimodal** traitant à la fois la voix, le texte et l’image du visage (via webcam) illustre également l’adaptabilité d’un réseau évolutif. À mesure que l’utilisateur se familiarise avec le système (introduction de nouveaux termes, évolution de ses habitudes gestuelles), les entités “voix” et “vision” ajustent leurs connexions et peuvent, le cas échéant, **former un cluster** spécialisé pour reconnaître l’utilisateur concerné. Les anciennes tâches (reconnaître d’autres utilisateurs) ne sont pas oubliées, car elles persistent dans des clusters distincts ; la connaissance acquise est ainsi **préservée** sur le long terme.

## Limites et défis de l’évolution continue

Si le réseau cherche à suivre de trop près les **fluctuations** de la synergie, il peut devenir **instable**, oscillant sans cesse ou détruisant prématurément des liens pertinents. Les paramètres  $\eta$ ,  $\tau$ ,  $\delta_{\text{fusion}}$ , etc., exigent donc un **calibrage** soigné pour maintenir un **équilibre** entre plasticité et stabilité (c’est le “dilemme stabilité-plasticité”).

Par ailleurs, dans un **modèle statique**, on fige les poids après l’apprentissage et on évalue la performance sur un ensemble de test. Dans le cadre d’un **réseau évolutif**, la performance est **susceptible de varier** au fil du temps, rendant nécessaire l’adoption de **métriques d’apprentissage continu** (p. ex. une mesure de l’erreur sur une fenêtre glissante).

Enfin, la reconfiguration permanente des liens implique un **coût de calcul**. Si le réseau compte  $n$  entités et que la synergie est évaluée entre toutes les paires, on fait face à  $O(n^2)$  opérations. Des stratégies de **parsimonie** (seuil de rupture, mise à jour partielle, échantillonnage) sont souvent indispensables pour demeurer **scalable** dans des systèmes de grande ampleur.

## Conclusion

L’**évolution dynamique** et l’**adaptation en continu** font du **Deep Synergy Learning** une forme de réseau “vivant”, capable de remodeler son organisation interne au fur et à mesure que l’environnement (ou les données) se transforment. Ce **changement de paradigme** – de l’entraînement ponctuel et figé à une auto-organisation permanente – ouvre la voie à des **applications** de l’IA plus flexibles et plus robustes, réduisant la dépendance à la supervision humaine et accroissant la **longévité** des modèles dans des scénarios réels et évolutifs.

Les sections suivantes (1.3.6 et 1.3.7) creuseront deux aspects cruciaux de cette perspective : la **robustesse** et la **résilience** que procure l'auto-organisation (1.3.6) et les **pistes** pour une approche plus générale de l'apprentissage (1.3.7), potentiellement orientée vers une IA plus proche de la cognition ou de l'**intelligence générale**.

### 1.3.6. Impacts sur la Robustesse et la Résilience des Modèles

L'une des conséquences les plus marquantes de l'**auto-organisation** et de la **synergie informationnelle** dans le **Deep Synergy Learning (DSL)** réside dans l'**amélioration** de la **robustesse** et de la **résilience** des systèmes d'apprentissage. Contrairement aux approches classiques, souvent vulnérables aux perturbations, aux changements de distribution ou aux défaillances partielles, le DSL tire parti de sa structure **évolutive** et **coopérative** pour mieux absorber ces aléas. Dans la présente section, nous examinons les notions de robustesse et de résilience, puis montrons comment elles s'expriment dans un réseau synergique.

#### 1.3.6.1. Définitions et enjeux

La **robustesse** d'un modèle désigne sa capacité à **maintenir** un niveau de performance élevé malgré la présence de **perturbations** ou d'**incertitudes**. Dans le contexte de l'IA, ces perturbations peuvent inclure :

- Du **bruit** dans les données (capteurs défectueux, images floues, signaux audio corrompus, etc.).
- Des **attaques adversariales** (petites perturbations “adversarial noise” rendant le modèle confus).
- Des **incohérences** ou **lacunes** dans les échantillons (valeurs manquantes, distribution très variée).

Un système robuste parvient à faire face à ces dégradations sans “collapsus” brutal de la performance.

La **résilience** va plus loin que la robustesse en impliquant non seulement la résistance aux perturbations, mais aussi la **capacité à récupérer** ou à **se réorganiser** après un choc. En IA, cela signifie que si la distribution des données change, ou si le système subit une panne partielle (certains capteurs tombent en panne, certains flux de données disparaissent), la résilience se manifeste par la **reconfiguration** interne permettant de continuer la tâche ou de s'en rapprocher au mieux.

#### 1.3.6.2. Mécanismes de robustesse dans un réseau synergique

Dans le DSL, plusieurs entités d'information peuvent recouvrir partiellement la même “zone de compétence” ou la même modalité, tout en se distinguant suffisamment pour apporter une **valeur ajoutée**. Du point de vue de la robustesse, cela signifie qu'un flux corrompu (ou une entité dysfonctionnelle) n'est pas catastrophique, car d'autres entités peuvent **prendre le relais**.

$$\omega_{i,j}(t) \rightarrow \omega_{k,j}(t+1), \quad \text{si } \mathcal{E}_i \text{ défaillante, alors } \mathcal{E}_k \text{ peut compenser.}$$

Lorsque la synergie entre une entité défectueuse  $\mathcal{E}_{\text{def}}$  et les autres entités chute (bruit, erreur répétée, etc.), les pondérations  $\omega_{\text{def},j}(t)$  s'affaiblissent progressivement. Le réseau “apprend” ainsi à **alléger** la connexion avec la source fautive et à **rediriger** l'information vers des entités plus fiables. Ce mécanisme évite un fort impact d'une entité isolée dysfonctionnelle.

Les entités réagissent localement au **bruit** ou aux **incohérences** en resserrant leurs liens synergiques avec celles qui restent cohérentes. On observe alors la création de **clusters** (voir 1.2.6) qui s'auto-assemblent autour de flux fiables ou complémentaires, renforçant la robustesse globale.

$$\mathcal{C}^* = \arg \max_{\mathcal{C}} \sum_{(i,j) \in \mathcal{C} \times \mathcal{C}} \omega_{i,j}, \quad \text{en excluant } \mathcal{E}_{\text{def}}.$$

#### 1.3.6.3. Résilience via l'adaptation en continu

Le DSL met en œuvre un apprentissage **continu** (section 1.3.5), ce qui signifie que le réseau s'ajuste aux nouvelles données ou aux nouveaux contextes sans nécessiter une refonte globale. Ainsi, si une modalité devient soudainement imprécise (caméra saturée de lumière, micro exposé à un fort bruit ambiant), le réseau réorganise ses **pondérations synergiques** pour ne plus dépendre de cette source.

$$\omega_{\text{camera}, \mathcal{E}_j}(t+1) = \omega_{\text{camera}, \mathcal{E}_j}(t) - \eta \left[ \tau \omega_{\text{camera}, \mathcal{E}_j}(t) \right] \quad \text{si } S(\mathcal{E}_{\text{camera}}, \mathcal{E}_j) \approx 0 \text{ (bruit élevé).}$$

La résilience suppose la **récupération** ou le maintien d’une performance acceptable même après un choc. Le fait de **supprimer** ou de **affaiblir** les liens inutiles (ou trompeurs) et de **renforcer** les liens pertinents génère une reconfiguration topologique :

$$G(t + 1) = \mathcal{U}[G(t), \{\omega_{i,j}\}, \text{nouveaux contextes}].$$

Ainsi, le réseau peut se redessiner sans qu’on doive recourir à une procédure hors-ligne complexe.

Les entités (et clusters) qui étaient **efficaces** pour d’anciennes tâches peuvent persister en parallèle, assurant une forme de **mémoire** ou de **transfert** vers des scénarios futurs. Cette cohabitation d’anciens et de nouveaux liens améliore la résilience, car le réseau ne jette pas systématiquement ses anciennes connaissances : il les conserve tant qu’elles restent synergiques, évitant le phénomène de “catastrophic forgetting”.

#### 1.3.6.4. Exemples d’Applications Robustes et Résilientes

Le **Deep Synergy Learning (DSL)** se révèle particulièrement adapté aux situations où la robustesse et la résilience constituent des critères essentiels. Deux scénarios illustrent la manière dont la **synergie informationnelle** et la **dynamique auto-organisée** du DSL permettent d’absorber des pannes, de réagir à des anomalies et de préserver la continuité opérationnelle. Dans chaque cas, la structuration coopérative du réseau offre une forme de **tolérance aux défaillances**, car l’affaiblissement d’un module ou d’un capteur ne remet pas en cause l’ensemble de la configuration. Les **exemples** ci-dessous mettent en évidence ces mécanismes de résilience et soulignent les **bienfaits** d’un réseau qui se **reconfigure** en temps réel.

##### A. Robotique et Systèmes Autonomes

Considérons un **robot** doté de multiples capteurs, tels que des caméras, un LIDAR, des gyroscopes, ainsi que d’autres senseurs spécialisés. Dans ce contexte, un modèle DSL peut absorber la défaillance partielle ou totale d’un capteur, sans qu’il soit nécessaire de procéder à un réapprentissage global. Cette propriété découle du caractère **auto-adaptatif** du Synergistic Connection Network (SCN) sous-jacent, où chaque capteur  $\mathcal{C}_i$  est traité comme une **entité** de l’ensemble, tandis que les liens  $\omega_{i,j}$  mesurent la **coopération** ou la **complémentarité** de deux capteurs  $i$  et  $j$ .

Dans le cas d’une **défaillance**, supposons que le capteur  $\mathcal{C}_k$  soit victime d’un bruit extrême ou d’une panne soudaine. Le **score de synergie**  $S(\mathcal{C}_k, \mathcal{C}_m)$  avec les autres capteurs  $\mathcal{C}_m$  diminuerait alors, ce qui provoquerait une mise à jour négative des liens  $\omega_{k,m}$  par l’équation

$$\omega_{k,m}(t + 1) = \omega_{k,m}(t) + \eta[S(\mathcal{C}_k, \mathcal{C}_m) - \tau \omega_{k,m}(t)],$$

où  $\eta$  désigne le taux d’apprentissage local et  $\tau$  le coefficient de régulation. À mesure que la **synergie** s’affaiblit, la pondération  $\omega_{k,m}$  décroît, de sorte que ce capteur devient quasi inactif au sein du SCN. Les autres capteurs, préservant des liens  $\omega$  élevés entre eux s’ils demeurent fiables et cohérents, continuent de coopérer pour fournir les informations nécessaires à la navigation ou à la réalisation des tâches robotiques. L’ajustement s’effectue en continu, sans qu’un entraînement complet soit relancé, montrant ainsi la **résilience** inhérente au DSL.

##### B. Surveillance Critique

Dans les domaines de la **surveillance** ou du **monitoring** (installations sensibles, applications médicales, détection d’intrusions), il est fréquent de recourir à un grand nombre de capteurs hétérogènes. Un système basé sur le DSL peut y déployer ses mécanismes d’**auto-organisation** de manière à détecter plus vite d’éventuelles anomalies et à tolérer les dysfonctionnements partiels d’un ou plusieurs capteurs. Le déploiement s’opère suivant la logique où chaque capteur  $\mathcal{D}_i$  constitue un **nœud** du SCN, et où le **score de synergie**  $S(\mathcal{D}_i, \mathcal{D}_j)$  quantifie la correspondance ou la complémentarité entre deux capteurs.

Lorsque l’un de ces nœuds présente une anomalie ou se déconnecte, la **pondération**  $\omega_{i,j}$  chute au fil des itérations, car la synergie mesurée devient faible. Mathématiquement, le processus de mise à jour

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

réduit progressivement les liens inutilisables ou compromis. La structure globale demeure néanmoins cohérente, car les capteurs restant fiables maintiennent des valeurs de  $\omega$  importantes, ce qui préserve la capacité du réseau à **fonctionner**. Le phénomène inverse se produit lorsqu’une alarme est déclenchée de façon synchronisée entre plusieurs capteurs. Dès lors que ceux-ci fournissent un signal convergent sur la présence d’un événement inhabituel, leur **synergie** augmente, et ils forment un **cluster** spécialisé dans l’analyse d’un risque potentiel. Cette coopérative rapide et flexible permet au SCN de hiérarchiser l’information critique. L’avantage repose sur l’**absence de réapprentissage intégral** et sur la capacité du système à s’**auto-réorganiser** pour mettre en avant les flux les plus pertinents.

##### Conclusion

Que ce soit dans un scénario de **robotique** multi-capteurs ou dans un dispositif de **surveillance** critique, le **Deep Synergy Learning** démontre ainsi son **aptitude** à gérer la panne ou la perte de fiabilité d’un sous-module sans perturber l’ensemble de l’architecture. La

dynamique de **mise à jour** des synergies et des liens  $\omega$  assure une continuité de service et un renforcement rapide des signaux valides, ce qui illustre la solidité opérationnelle et l'efficacité de cette **approche auto-organisée** pour la gestion des situations critiques ou complexes.

### 1.3.7. Perspectives pour une Approche plus Globale de l'Apprentissage

Les sections précédentes (1.3.1 à 1.3.6) ont mis en évidence l'importance de l'**auto-organisation** dans le Deep Synergy Learning (DSL), ses fondements biologiques et cognitifs (1.3.1), les concepts d'émergence et de feedback (1.3.2), la comparaison avec les méthodes classiques (1.3.3), le rôle crucial de la multi-modalité (1.3.4), la dynamique adaptative (1.3.5) et l'impact sur la robustesse/résilience (1.3.6). Nous en arrivons maintenant à la question centrale : **dans quelle mesure** cette architecture auto-organisée peut-elle contribuer à une **approche plus globale** de l'apprentissage, allant potentiellement vers une IA plus générale, moins cloisonnée et plus proche de la cognition humaine ?

#### 1.3.7.1. Vers une intégration de multiples paradigmes

L'un des attraits majeurs du DSL réside dans sa **flexibilité** structurelle, permettant l'intégration simultanée de tâches variées (reconnaissance d'images, traitement du langage, analyse de signaux). Plutôt que de multiplier les sous-modules indépendants, le DSL **mutualise** la capacité d'adaptation et de co-évolution. Cela ouvre la voie à des systèmes capables d'aborder des **problèmes multi-domaines** de façon unifiée.

Beaucoup de chercheurs s'interrogent sur la possibilité de combiner une **représentation symbolique** (logique, règles, ontologies) et des **réseaux neuronaux** (approche sub-symbolique). Le DSL offre un espace où des **entités symboliques** (représentant des concepts, des règles) pourraient coexister avec des **entités sub-symboliques** (features non supervisées, clusters contextuels), et forger des **liens synergiques** si ces représentations s'avèrent mutuellement bénéfiques.

$$\omega_{\text{symbolique,sub-symbolique}}(t+1) = \omega_{\text{symbolique,sub-symbolique}}(t)\eta[S(\mathcal{E}_{\text{symbolique}}, \mathcal{E}_{\text{sub-symbolique}})].$$

Une autre piste consiste à inclure des **boucles de récompense** (façon apprentissage par renforcement) à l'intérieur du DSL, de manière à ce que la **synergie** prenne en compte non seulement l'interaction entre entités, mais aussi un **signal de performance** plus global. Cela pourrait mener à des architectures où l'**exploration** et la **sélection** des connexions synergiques s'effectuent au service d'une stratégie d'agent, par exemple dans un environnement en évolution (robotique, jeux, etc.).

#### 1.3.7.2. Apprentissage contextuel et raisonnement adaptatif

Dans de nombreux problèmes (analyse de scène, dialogue, diagnostic médical), l'**environnement** ou la **situation** évolue. Le DSL permet de faire émerger des **clusters contextuels**, regroupant les entités les plus pertinentes pour un contexte donné. Ces clusters peuvent ensuite se **désagréger** ou se **recombinaer** quand le contexte change, offrant un niveau de **contextualisation dynamique** qu'on retrouve rarement dans les modèles hiérarchiques figés.

Plutôt que de raisonner par règle ("si A alors B"), le réseau peut **découvrir** que certaines combinaisons d'entités  $\{\mathcal{E}_i, \mathcal{E}_j, \dots\}$  produisent un **effet global** pertinent. On pourrait formaliser cette découverte comme un **raisonnement émergent**, où l'apparition d'un cluster synergique équivaut à la création d'un "concept" ou d'une "hypothèse" confirmée par l'amélioration de la performance ou de la cohérence.

$$\text{Concept}_\alpha \leftrightarrow \{\mathcal{E}_i, \mathcal{E}_j, \dots\} \quad \text{avec} \quad \sum_{(i,j) \in \alpha} \omega_{i,j} > \theta.$$

Les entités dans le DSL peuvent porter un **état interne** (1.2.1), ce qui permet de **mémoriser** certains événements, associations ou transitions. Couplé à l'idée de clusters, on obtient une forme de mémoire **distribuée** et **adaptative**, capable de se ré-agencer au fil du temps, au lieu d'être confinée dans des architectures rigides (ex. LSTM).

#### 1.3.7.3. Ouverture vers une IA plus proche de la cognition biologique

Nous avons déjà souligné les parallèles entre la synapse biologique et les connexions synergiques (sections 1.3.1, 1.3.2). Un système auto-organisé qui évolue au fil du temps, fusionne ou dissocie des entités, renforce ou supprime des liens, **s'approche** d'une **dynamique neuronale** élémentaire. Il est donc envisageable que des **architectures DSL** se montrent plus aptes à **simuler** ou **comprendre** certains phénomènes cognitifs.

Bien que spéculative, la question d’une **conscience artificielle** ou d’une **auto-consistance** cognitive pourrait trouver dans le DSL un champ d’exploration. Certains travaux suggèrent qu’un degré d’**intégration d’information** (comme l’approche de la “Phi measure” de Giulio Tononi) est requis pour émerger une forme de conscience. Si le DSL parvient à développer des **clusters** hautement connectés et persistants (au sens “informationnellement intégrés”), il pourrait servir de terrain expérimental pour avancer sur ces hypothèses.

L’IA forte (IAG) ou Intelligence Artificielle Générale se définit par la capacité d’un système à maîtriser un large éventail de tâches, à apprendre de manière autonome et à faire face à des environnements variés. La **plasticité** et la **co-évolution** du DSL, sa manière de laisser chaque entité s’adapter, fusionner, se spécialiser, suggèrent un potentiel pour **surmonter** les obstacles auxquels se heurtent les méthodes rigides (catastrophic forgetting, absence de transfert, etc.). Cela ne garantit pas l’émergence d’une IAG, mais en offre une piste conceptuelle plus proche du fonctionnement adaptatif du vivant.

#### 1.3.7.4. Principaux défis et voies de recherche

Le **Deep Synergy Learning (DSL)** introduit un cadre inédit, reposant sur une **interaction dynamique** et une **auto-organisation** continue des entités d’information. Malgré l’attrait de cette proposition, il subsiste plusieurs défis mathématiques, algorithmiques et pratiques, dont la résolution s’avère essentielle pour la généralisation à grande échelle. Sur le plan **computationnel**, l’évaluation des synergies entre un nombre important d’entités peut impliquer une complexité en  $O(n^2)$ , où  $n$  désigne le nombre d’entités. Cette explosion combinatoire requiert des stratégies de **parsimonie** et de **sparsification** afin de restreindre la croissance du nombre de connexions à évaluer. Des techniques d’**échantillonnage** adaptatif peuvent aussi être envisagées, de manière à ne considérer que les paires d’entités présentant une synergie potentiellement élevée. D’un point de vue plus formel, on peut réduire la densité des connexions actives en imposant un mécanisme de seuil sur les poids  $\omega_{ij}$ , de sorte qu’une liaison  $(i, j)$  ne soit conservée que si

$$\omega_{ij} > \omega_{\min},$$

avec  $\omega_{\min}$  un paramètre choisi pour garantir la structure globale du réseau.

S’agissant de l’**interprétabilité** et de l’**explicabilité**, la capacité du DSL à faire émerger des **clusters** et à redessiner en continu ses connexions favorise une organisation plus fluide, mais rend ardue l’analyse a posteriori des décisions. Pour éclairer la nature d’une prédiction, il devient nécessaire de retracer l’historique des synergies et d’examiner l’état interne des entités concernées. La mise à jour de chaque poids  $\omega_{ij}$  peut être décrite par une équation de type

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

où  $\eta$  désigne le taux d’apprentissage et  $\tau$  le coefficient de décroissance. Cette dynamique rend possible l’apparition ou la suppression de liens dont la contribution n’est plus jugée pertinente, tout en renforçant ceux qui semblent les plus profitables au regard de la synergie mesurée. Cependant, un tel fonctionnement complique la traçabilité : pour extraire des règles ou des justifications, il faudrait stocker ou synthétiser l’ensemble des états successifs, ainsi que les évaluations de  $S(\mathcal{E}_i, \mathcal{E}_j)$ .

La question de la **convergence** suscite également un vif intérêt. L’étude de la stabilité et des attracteurs potentiels liés aux mises à jour des poids dans le cadre d’un **système dynamique** exige d’analyser l’évolution de l’**énergie** du réseau ou d’une **fonction de Lyapunov** associée. Si l’on considère une énergie

$$E(t) = - \sum_{i \neq j} \omega_{ij}(t) S(\mathcal{E}_i, \mathcal{E}_j),$$

la décroissance de  $E(t)$  au fil des itérations peut servir d’indicateur de convergence vers un état stable où les pondérations cessent d’évoluer de manière significative. L’identification de conditions garantissant l’existence d’un minimum global ou local, ainsi que la caractérisation de la vitesse de convergence, demeurent des questions ouvertes pour la communauté scientifique. La présence de boucles de rétroaction complexes, de reconfigurations topologiques et d’interactions non linéaires peut engendrer des comportements oscillatoires, voire chaotiques, si aucune **régularisation** adéquate n’est mise en place.

Dans les **domaines critiques** (santé, finance, transport, sécurité), le DSL pourrait apporter une meilleure **robustesse** grâce à la **fusion multimodale** et à la **répartition adaptative** des informations, permettant de gérer des données hétérogènes ou partiellement manquantes. Toutefois, de telles applications exigent des garanties fortes sur la fiabilité et la sûreté de fonctionnement, impliquant des **processus de certification** exigeants. Il devient essentiel d’y intégrer des mécanismes de **sécurité** contre d’éventuelles attaques adversariales, où de subtiles perturbations de l’entrée peuvent tromper même des systèmes neuronaux performants. La dynamique adaptative et coopérative du DSL pourrait, d’un côté, renforcer la **résilience** en détectant des incohérences via les synergies internes, mais de l’autre côté, elle complique l’établissement de **bornes** de sûreté ou de **preuves** formelles de robustesse.

Ainsi, malgré un potentiel considérable, le DSL fait face à des enjeux théoriques et pratiques déterminants pour son adoption à grande échelle. La **réduction de la complexité algorithmique**, l’**amélioration de l’interprétabilité**, l’**étude minutieuse de la convergence** et l’**adaptation aux environnements critiques** constituent autant de pistes de recherche centrales pour consolider ce paradigme. Les avancées futures consisteront notamment à concevoir des **stratégies d’approximation** limitées en ressources de calcul, à développer des **méthodes d’exploration** et de **visualisation** capables d’exposer la structure émergente du réseau, et à établir des **cadres**

**mathématiques** plus complets validant la stabilité et la fiabilité du processus d'**auto-organisation**. L'ensemble de ces travaux pourra contribuer à faire du Deep Synergy Learning un jalon clé dans l'évolution de l'IA vers une plus grande **plasticité, résilience et intelligence**.

#### 1.3.7.5. Conclusion : vers un nouveau paradigme d'apprentissage

En somme, l'auto-organisation et la synergie, éléments centraux du **Deep Synergy Learning**, ouvrent la voie à une **approche plus globale** de l'apprentissage, susceptible d'englober plusieurs paradigmes existants (supervisé, non supervisé, renforcement, symbolique, etc.) tout en proposant des mécanismes d'évolution continue, de raisonnement contextuel et de mémorisation distribuée.

Cette vision n'est pas encore un aboutissement : de nombreux **défis** scientifiques, méthodologiques et éthiques (scalabilité, interprétabilité, robustesse, etc.) demeurent à relever. Néanmoins, le DSL fournit d'ores et déjà un **cadre unificateur** pour penser l'**IA adaptative**, apte à accueillir de multiples flux d'information et à co-évoluer de façon organique, en s'inspirant des systèmes vivants.

Les chapitres suivants approfondiront encore la **formulation mathématique** du DSL, ses **algorithmes** et ses **applications pratiques**, en vue de concrétiser ce nouveau paradigme et d'explorer son potentiel pour des systèmes intelligents, au-delà des limites imposées par les architectures hiérarchiques classiques.

### Problème 1 : Analyse de la Dynamique des Pondérations Synergiques

On considère un réseau de  $n$  entités  $\{\mathcal{E}_1, \dots, \mathcal{E}_n\}$ . Les pondérations synergiques  $\omega_{i,j}(t)$  évoluent selon la règle suivante :

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

où  $\eta > 0$  et  $\tau > 0$  sont des constantes, et  $S(\mathcal{E}_i, \mathcal{E}_j)$  représente une mesure (constante) de synergie entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ .

#### 3. Équilibre local

Montrez qu'à temps long, pour chaque paire  $(i, j)$ , la pondération  $\omega_{i,j}(t)$  peut atteindre un point d'équilibre  $\omega_{i,j}^*$ . Exprimez  $\omega_{i,j}^*$  en fonction de  $S(\mathcal{E}_i, \mathcal{E}_j)$ ,  $\eta$  et  $\tau$ .

#### 4. Stabilité linéaire

Établissez la condition (en termes de  $\eta$  et  $\tau$ ) sous laquelle l'équilibre  $\omega_{i,j}^*$  est **stable** dans le cadre d'une analyse linéaire (linéarisation autour de  $\omega_{i,j}^*$ ).

#### 5. Convergence globale

Proposez une justification ou un argument (non nécessairement exhaustif) permettant d'expliquer la **convergence** globale du système  $\{\omega_{i,j}(t)\}$  vers un ensemble d'états d'équilibre, en supposant que tous les couples  $(i, j)$  obéissent à la même loi de mise à jour.

#### 6. Impact de la régularisation

Supposons qu'on introduise un terme quadratique supplémentaire dans la mise à jour :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S_{i,j} - \tau \omega_{i,j}(t)] - \alpha \omega_{i,j}(t)^3,$$

avec  $\alpha > 0$ . Discutez l'impact potentiel de ce terme "cubic" sur la stabilité et la répartition finale des  $\omega_{i,j}(t)$ . Quelles questions mathématiques (bifurcations, multiplicité des équilibres) cela soulève-t-il ?

#### 7. Extension n-aire

Généralisez la dynamique ci-dessus à une **synergie n-aire**,  $S(\mathcal{E}_{i_1}, \dots, \mathcal{E}_{i_n})$ , où la mise à jour de  $\omega_{i_1, \dots, i_n}(t)$  dépend d'un terme commun  $\tau \omega_{i_1, \dots, i_n}(t)$ . Proposez une formulation des équations et discutez le principal défi analytique pour montrer la convergence dans ce cas.

### Problème 2 : Émergence de Clusters et Attracteurs de Graphe

Soit un graphe  $G(t)$  dont les sommets sont les entités  $\{\mathcal{E}_1, \dots, \mathcal{E}_n\}$  et les arêtes sont pondérées par  $\omega_{i,j}(t)$ . On se propose d'étudier l'**émergence** de sous-graphes fortement connectés (clusters) en tant qu'attracteurs de la dynamique.

#### 8. Fonction de coût globale

Introduisez une fonction "énergie" ou "coût" globale :

$$\mathcal{J}(G(t)) = - \sum_{(i,j)} \omega_{i,j}(t) S_{i,j} + \beta \sum_{(i,j)} [\omega_{i,j}(t)]^2,$$

où  $S_{i,j}$  est la synergie (positive ou négative) et  $\beta > 0$  un paramètre de régularisation. Expliquez pourquoi minimiser  $\mathcal{J}$  favorise la **création de clusters** de liens positifs (forte synergie) et la **suppression** de liens inutiles.

#### 9. État stable et clusterisation

Montrez qu'un **état stable** (un minimum local de  $\mathcal{J}$ ) peut correspondre à la formation d'un sous-ensemble  $\mathcal{C}$  tel que  $\omega_{i,j}$  est élevé si  $i, j \in \mathcal{C}$ , et faible sinon. Expliquez, sur un plan mathématique, comment on peut interpréter  $\mathcal{C}$  comme un **cluster**.

#### 10. Analyse de la multiplicité



Justifiez l'existence possible de **plusieurs** configurations d'équilibre local, chacune correspondant à un partitionnement différent du graphe. Quels outils d'optimisation ou de théorie des graphes peut-on invoquer pour analyser la multiplicité de ces minima ?

#### 11. Transitions entre attracteurs

Décrivez un scénario mathématique permettant à un **sous-ensemble de sommets**  $\mathcal{C}$  de se dissocier en deux clusters  $\mathcal{C}_1$  et  $\mathcal{C}_2$ . Quelle condition sur la somme des synergies (interne vs. externe) pourrait induire la scission ? Dans quel cas obtient-on une fusion ?

### Problème 3 : Analyse de la Stabilité-Plasticité dans l'Auto-Organisation

On considère la double contrainte "stabilité-plasticité" : le réseau doit être suffisamment **plastique** pour intégrer de nouveaux signaux, mais suffisamment **stable** pour ne pas détruire systématiquement les acquis précédents.

#### 12. Formulation d'un critère mathématique

Proposez une formalisation où la "stabilité"  $\mathcal{S}$  et la "plasticité"  $\mathcal{P}$  sont deux fonctions mesurant respectivement la conservation des liens existants et la facilité de créer/supprimer des liens. Définissez un **objectif**  $\Gamma(\mathcal{S}, \mathcal{P})$  à maximiser et discutez l'arbitrage mathématique qui en découle.

#### 13. Stabilité locale vs. stabilité globale

Montrez comment on peut distinguer **stabilité locale** (analogue à la dérivée de  $\omega_{i,j}$  près d'un équilibre) et **stabilité globale** (résilience à de grandes perturbations). Proposez un schéma d'analyse linéaire pour la stabilité locale des pondérations, et un schéma plus qualitatif pour la stabilité globale.

#### 14. Équation couplée avec un taux de plasticité

Introduisez un **taux de plasticité**  $\alpha(t)$  qui peut varier dans le temps :

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

Identifiez les conditions sur  $\alpha(t)$  pour que le réseau n'entre pas dans des oscillations permanentes.

#### 15. Compromis entre exploitation et exploration

Interprétez le rôle de  $\alpha(t)$  comme un **compromis** entre "exploitation" (stabilité) et "exploration" (plasticité). Montrez, par un raisonnement mathématique ou par une étude de fonctions de Lyapunov, qu'un **décroissement progressif** de  $\alpha(t)$  peut favoriser la convergence vers un cluster stable à long terme.

### Problème 4 : Modélisation de la Multi-modalité et de la Synergie Conditionnelle

On souhaite étudier un système où chaque entité  $\mathcal{E}_i$  appartient à l'une de plusieurs **modalités** (par exemple : visuel, auditif, textuel). La synergie entre deux entités dépend aussi d'un **contexte**  $\mathbf{c}$ .

#### 16. Synergie conditionnelle

Formalisez la mesure :

$$S(\mathcal{E}_i, \mathcal{E}_j | \mathbf{c}),$$

où  $\mathbf{c} \in \mathcal{C}$  est une variable contextuelle. Quels axiomes mathématiques (monotonie, symétrie partielle, etc.) peut-on imposer à  $S(\cdot | \mathbf{c})$  pour qu'elle reste cohérente ?

#### 17. Mise à jour des pondérations selon le contexte

Proposez une loi d'évolution contextuelle :

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

Montrez comment cette équation peut, en pratique, conduire à la formation de **clusters spécifiques** à certains contextes  $\mathbf{c}$ .

### 18. Transfert de contexte

Supposons que le contexte  $\mathbf{c}_1$  soit “nuit” et  $\mathbf{c}_2$  soit “jour”. Étudiez la possibilité d’un **transfert** de liens synergiques depuis  $\mathbf{c}_1$  vers  $\mathbf{c}_2$  si certaines entités se révèlent robustes aux deux contextes. Quel est le rôle d’une éventuelle **intersection** d’ensembles de clusters  $\mathcal{C}_1 \cap \mathcal{C}_2 \neq \emptyset$  ?

### 19. Extension à la multimodalité n-aire

Discutez comment généraliser la synergie binaire à la synergie n-aire conditionnelle,  $S(\mathcal{E}_{i_1}, \dots, \mathcal{E}_{i_n} \mid \mathbf{c})$ . Quels problèmes de **complexité** surgissent si l’on veut mettre à jour l’ensemble  $\omega_{i_1, \dots, i_n}(t)$  en fonction d’un contexte  $\mathbf{c}(t)$  ?

## Problème 5 : Étude Formelle de l’Émergence et de la Cognition Distribuée

Ce dernier problème se concentre sur la **dimension cognitive** et les possibilités d’aller vers une **approche globale** de l’apprentissage, inspirée par la section 1.3.7.

### 20. Assemblées auto-organisées et “concepts”

Modélisez mathématiquement la notion de “concept émergent” comme un **cluster**  $\mathcal{C}$  dont la somme des pondérations internes dépasse un seuil  $\theta$ . Proposez une définition fonctionnelle :

$$\text{Concept}_{\alpha} := \left\{ \mathcal{E}_i \mid \sum_{j \in \alpha} \omega_{i,j} > \theta \right\}.$$

Comment prouver que ce concept est stable sous l’évolution des pondérations ?

### 21. Hiérarchie de concepts

Envisagez la construction d’une **hiérarchie** de concepts  $\mathcal{H}$ , où un “métaconcept”  $\alpha^+$  apparaît si deux concepts  $\alpha_1$  et  $\alpha_2$  ont une forte synergie inter-clusters. Formulez les conditions nécessaires :

$$\sum_{(i \in \alpha_1, j \in \alpha_2)} \omega_{i,j} > \theta_2,$$

et discutez la formation d’une structure arborescente ou en treillis.

### 22. Mémoire distribuée et clusters persistants

Définissez une **fonction de persistance** mesurant la durée pendant laquelle un cluster  $\mathcal{C}$  reste cohérent (c.-à-d. affiche des pondérations internes supérieures à un certain seuil). Prouvez ou argumentez qu’une entité ayant participé longtemps à un cluster  $\mathcal{C}$  garde une “mémoire partielle” de cette association, même si  $\omega_{i,j}$  se réduit par la suite.

### 23. Discussion sur la convergence vers une IA forte

Proposez un **modèle** ou un **schéma** montrant comment ces mécanismes (formation de concepts, hiérarchie, mémoire persistante) peuvent contribuer à une forme plus globale de “cognition distribuée”. Formulez au moins deux questions mathématiques ouvertes (liées aux attracteurs, à la complexité, ou à la dynamique asymptotique) qui se posent si l’on veut prouver la “capacité générale” d’un tel réseau.

### 24. Lien avec la théorie de l’information intégrée

(Optionnel) Évoquez le lien potentiel entre la somme de synergies  $\sum_{i,j} \omega_{i,j} S(\mathcal{E}_i, \mathcal{E}_j)$  et une mesure d’**information intégrée**. Quels prolongements théoriques imaginez-vous pour formaliser la “quantité d’intégration” qui pourrait soutenir une cognition plus “globale” ?

Ces **cinq problèmes** offrent un panorama de questionnements mathématiques autour de la **section 1.3** (Importance de l’Auto-Organisation) : la **dynamique** des pondérations (Problème 1), l’**émergence** et la **clusterisation** (Problème 2), l’équilibre subtil **stabilité-plasticité** (Problème 3), la gestion de la **multi-modalité** et du **contexte** (Problème 4), et enfin une ouverture vers la **cognition distribuée** (Problème 5). Aucun calcul numérique n’est exigé ; il s’agit exclusivement de **questions formelles**, destinées à approfondir la compréhension mathématique de l’auto-organisation dans le cadre du **Deep Synergy Learning**.

## Étude mathématique du système dynamique et résolution de l'équation d'équilibre

L'équation donnée :

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

représente un système dynamique discret, où les pondérations  $\omega_{i,j}(t)$  évoluent sous l'influence des synergies  $S(\mathcal{E}_i, \mathcal{E}_j)$ , du taux d'apprentissage  $\eta > 0$ , et du coefficient de décroissance  $\tau > 0$ .

### 1. Condition d'équilibre

À l'équilibre, les pondérations cessent de changer, ce qui implique que :

$$\omega_{i,j}(t+1) - \omega_{i,j}(t) = 0.$$

Substituons cette condition dans l'équation dynamique :

$$0 = \eta[S(\mathcal{E}_i, \mathcal{E}_j) - \tau\omega_{i,j}(t)].$$

En simplifiant, on obtient :

$$\omega_{i,j}^* = \frac{S(\mathcal{E}_i, \mathcal{E}_j)}{\tau},$$

où  $\omega_{i,j}^*$  représente la pondération à l'équilibre.

### 2. Stabilité de l'équilibre

Pour étudier la **stabilité** de cet équilibre, nous examinons comment une perturbation autour de  $\omega_{i,j}^*$  évolue.

#### a) Écart autour de l'équilibre

Posons :

$$\delta\omega_{i,j}(t) = \omega_{i,j}(t) - \omega_{i,j}^*.$$

En remplaçant  $\omega_{i,j}(t)$  dans l'équation dynamique :

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

Cela se simplifie en :

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

#### b) Condition de stabilité

Pour que  $\delta\omega_{i,j}(t) \rightarrow 0$  (stabilité), il faut que :

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

Cela donne :

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

Ainsi, le système est **stable** si le produit  $\eta\tau$  est dans l'intervalle  $]0,2[$ .

### 3. Cas des oscillations

Si  $\eta\tau$  dépasse 2, le facteur  $1 - \eta\tau$  devient négatif, ce qui entraîne une **oscillation** de  $\delta\omega_{i,j}(t)$  autour de l'équilibre. Cela correspond à un comportement oscillatoire amorti ou amplifié selon les paramètres.

#### *Oscillations amorties*

Si  $\eta\tau > 2$  mais reste modéré, les oscillations se réduisent progressivement et finissent par converger.

#### *Oscillations non amorties*

Si  $\eta\tau \gg 2$ , les oscillations peuvent persister indéfiniment ou même s'amplifier, menant à un régime instable.

### 4. Analyse globale avec la Jacobienne

La Jacobienne du système au point d'équilibre  $\Omega^* = [\omega_{i,j}^*]$  s'écrit :

$$J_{i,j} = \frac{\partial}{\partial \omega_{k,l}} [\omega_{i,j}(t+1) - \omega_{i,j}(t)].$$

En calculant explicitement :

$$\frac{\partial}{\partial \omega_{k,l}} [\omega_{i,j}(t+1) - \omega_{i,j}(t)] = \begin{cases} -\eta\tau, & \text{si } (i,j) = (k,l), \\ 0, & \text{sinon.} \end{cases}$$

La Jacobienne est donc diagonale avec les termes  $-\eta\tau$  sur la diagonale. La stabilité globale dépend des valeurs propres, qui sont  $1 - \eta\tau$ . Si toutes ces valeurs propres vérifient  $|1 - \eta\tau| < 1$ , alors le système est stable.

### 5. Simulation numérique (résolution discrète)

#### *Paramètres de simulation :*

- Nombre de nœuds :  $N = 10$ ,
- Synergies  $S(i,j)$  : semi-aléatoires ( $\sim \mathcal{U}(0,1)$ ),
- Taux d'apprentissage  $\eta = 0.1$ ,
- Coefficient de décroissance  $\tau = 1.0$ .

#### *Algorithme :*

25. Initialiser  $\omega_{i,j}(0) \sim \mathcal{U}(0,0.01)$ ,
26. Mettre à jour  $\omega_{i,j}(t)$  à chaque itération selon l'équation dynamique,
27. Calculer  $\delta\omega_{i,j}(t)$  pour vérifier la convergence.

#### *Résultats attendus :*

28. Convergence rapide vers  $\omega_{i,j}^* \approx S(i,j)/\tau$  si  $\eta\tau < 2$ ,
29. Oscillations visibles si  $\eta\tau \geq 2$ .

### 6. Applications et interprétation

- **Réseaux biologiques** : Modélisation des interactions dynamiques entre neurones ou molécules.
- **Systèmes physiques** : Étude des attracteurs dans les systèmes oscillants ou auto-organisés.
- **Réseaux artificiels** : Régulation des pondérations dans un réseau neuronal ou un DSL (Deep Synergy Learning).

## Conclusion

L'étude de l'équation  $\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta[S(E_i, E_j) - \tau\omega_{i,j}(t)]$  met en lumière des dynamiques d'auto-organisation riches et complexes. Elle montre comment les synergies et les paramètres de régulation influencent la stabilité et les comportements oscillatoires, offrant des outils puissants pour comprendre et modéliser des systèmes complexes.

#### 1.4. Architecture Générale du DSL

Dans les chapitres précédents, nous avons défini les **fondements conceptuels** du Deep Synergy Learning (DSL), en soulignant notamment l'importance de l'**auto-organisation** et des **synergies** entre entités d'information (sections 1.2 et 1.3). Nous savons désormais que chaque entité n'est pas qu'un simple vecteur de données : elle peut évoluer, interagir, et créer de nouvelles représentations lorsqu'elle coopère avec d'autres entités. Ces processus de coopération ne sont pas imposés par un schéma hiérarchique rigide, mais émergent librement à mesure que les entités détectent un **gain** dans leur association.

Afin de concrétiser ces principes dans un **cadre unifié**, il est nécessaire de décrire l'**architecture générale** du DSL. Autrement dit, comment ces entités s'organisent-elles ? Quelles structures se dégagent quand la synergie est élevée entre certaines paires (ou ensembles) d'entités ? Et comment modéliser leur évolution au fil du temps ? La section 1.4 répond à ces questions en introduisant :

- Les **principes de base** des entités et des liens (1.4.1)
- La notion de **Synergistic Connection Network (SCN)** (1.4.2)
- La formation de **clusters** et de **macro-clusters** (1.4.3)
- Les différentes **fonctions de synergie** (1.4.4)
- Le fonctionnement **adaptatif** des pondérations (1.4.5)
- Les interactions **directes** et **indirectes** (1.4.6)
- Les cas particuliers de **synergie binaire** et **n-aire** (1.4.7)

C'est cette architecture globale du **DSL** qui soutient la plasticité et la capacité d'adaptation du système : plutôt que de transmettre passivement les données dans des couches, chaque entité peut, en temps réel, renforcer ou affaiblir ses liens avec d'autres entités, créer des regroupements spontanés (clusters), et faire émerger de nouvelles représentations plus riches que la somme de leurs composantes individuelles.

##### 1.4.1. Principe de Base : Entités et Liens Synergiques

Le **Deep Synergy Learning** envisage chaque **entité d'information** comme un **nœud actif** de l'architecture globale. Plutôt que de propager passivement un signal, comme le ferait un neurone dans un réseau hiérarchique, cette entité adopte un rôle décisif en disposant de **paramètres internes**  $\theta_k$ , souvent constitués d'une **représentation**  $\mathbf{x}_k$  (vectorielle ou tensorielle), d'un **état interne**  $\mathbf{s}_k$  qui se modifie au gré des interactions, et de tout **hyperparamètre** requis pour la modalité concernée (par exemple un encodeur audio ou un embedding textuel). Chaque entité s'accompagne également de **mécanismes de décision** qui lui permettent d'**observer** la synergie obtenue lorsqu'elle se connecte à d'autres entités et, selon cette observation, de renforcer, d'affaiblir ou de rompre certains liens. Elle peut même, dans les cas où la coopération se révèle durablement élevée, envisager une **fusion** avec une autre entité ou, au contraire, une **spécialisation** en segmentant ses propres paramètres pour répondre à des tâches spécifiques. Cette description, encore préliminaire, fait déjà ressortir l'idée que les **nœuds** du DSL sont bien plus dynamiques que dans un réseau de neurones classique, puisque chacun "recherche" les coopérations les plus fructueuses pour améliorer sa représentation tout en contribuant à la robustesse du réseau dans son ensemble.

Chaque entité  $\mathcal{E}_i$  peut se connecter à une autre entité  $\mathcal{E}_j$  par le biais d'une **pondération synergique**  $\omega_{i,j}(t)$ , laquelle traduit la **valeur ajoutée** perçue dans cette association. Quand la synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$  est forte, on observe un accroissement de  $\omega_{i,j}(t)$ , tandis que dans le cas contraire, ce lien tend à se résorber puis à disparaître. L'évolution de  $\omega_{i,j}$  se modélise alors par une équation simple, par exemple

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

où  $\eta$  désigne le taux d'apprentissage et  $\tau$  un paramètre de régularisation. Cette formulation, exposée plus en profondeur dans les sections ultérieures, peut aussi s'enrichir de seuils pour limiter la croissance des poids ou supprimer les connexions trop faibles, assurant de la sorte une **régulation** de la densité du graphe. Une fois que l'on modélise ainsi les **liens synergiques**, on se dote d'une **matrice d'adjacence** évolutive qui dicte quels nœuds interagissent, quand ils le font et avec quelle intensité.

La grande différence avec une organisation hiérarchique est que l'information n'est plus obligée de transiter selon des voies prédéfinies. Toute entité peut potentiellement échanger avec n'importe quelle autre, pour peu que la **pondération synergique** atteigne un niveau jugé satisfaisant. Dans le cadre d'une tâche multimodale, il est donc envisageable que des entités "visuelles" se lient directement à des entités "textuelles" si elles y trouvent un bénéfice mutuel. Il se forme alors des **clusters** éphémères ou plus durables, réunissant deux, trois ou davantage d'entités, et la persistance de ces regroupements dépend de la permanence de leur synergie. De tels micro-réseaux se constituent et se défont librement, donnant lieu à une **auto-organisation** dont l'architecture n'a pas été spécifiée a priori mais découle de la dynamique interne du système.

Cette perspective se décrit volontiers comme un **graphe évolutif**  $G(t)$ . Les **nœuds** correspondent aux entités  $\{\mathcal{E}_1, \dots, \mathcal{E}_n\}$ , et les **arêtes** représentent les liaisons  $\omega_{i,j}(t)$ . À chaque itération, une règle d'actualisation  $\mathcal{U}$  recalcule  $\omega_{i,j}(t+1)$  en tenant compte de la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  et de la **régularisation** (qui inclut, au besoin, la suppression des liens faibles). Cela rend possibles des analyses croisées avec la **théorie des graphes** ou la **dynamique non linéaire**, et l'on peut même faire appel à des méthodologies d'**optimisation combinatoire** lorsqu'on cherche, par exemple, à minimiser un coût global ou à identifier une configuration de connexions correspondant à un optimum local.

Cette approche basée sur des **nœuds actifs** et des **arêtes dynamiques** confère au **DSL** une **flexibilité** qui se révèle supérieure à celle de nombreuses architectures traditionnelles. La topologie n'est plus figée dans des couches de traitement ; elle se réorganise au gré des opportunités de synergie détectées, ce qui est d'un intérêt crucial pour les situations multimodales ou pour l'adaptation à des données en évolution. Les entités elles-mêmes peuvent fusionner, se subdiviser ou changer de paramétrage, et toute la structure s'oriente vers une organisation où la synergie la plus forte est mise en avant. Cette logique favorise l'apparition de **propriétés émergentes** comme la consolidation de **macro-clusters** ou l'élaboration de schémas coopératifs inédits, sans qu'aucune hiérarchie ne soit imposée dès le départ.

En somme, la clef de voûte du **DSL** réside dans l'idée que chaque entité constitue un **nœud autonome**, et chaque connexion, un **lien synergique** dont l'évolution est régie par le degré d'enrichissement mutuel. Les observations de clusters, de reconnections ou de regroupements imprévus y sont donc la norme, et ce caractère **distribué** et **adaptatif** positionne le DSL comme un prolongement original des paradigmes existants, taillé pour gérer la variété et la complexité croissantes des données actuelles.

#### 1.4.2. Présentation du Synergistic Connection Network (SCN)

Le **Synergistic Connection Network (SCN)** constitue la **pièce maîtresse** du Deep Synergy Learning (DSL) en matérialisant à la fois l'**espace** dans lequel les entités évoluent et le **mécanisme** même qui autorise la naissance, la transformation ou la disparition de leurs liens. Alors que les réseaux de neurones classiques reposent sur une **topologie fixée** avant l'entraînement, le SCN propose, au contraire, une **structure adaptive** dont les entités et leurs connexions se reconfigurent de manière autonome au fil du temps.

La représentation du SCN comme **graphe** place les **entités**  $\{\mathcal{E}_1, \mathcal{E}_2, \dots, \mathcal{E}_n\}$  en tant que **nœuds**, connectés par des **pondérations synergiques**  $\omega_{i,j}$  capables d'évoluer : chaque entité correspond à un bloc fonctionnel (par exemple visuel, textuel ou auditif) et peut, en fonction de la **synergie** détectée, renforcer ou, au contraire, réduire sa liaison avec une autre entité. Cette plasticité confère au SCN un caractère **vivant** : la structure n'est jamais figée, mais se **recompose** continuellement, en conservant les connexions jugées utiles et en éliminant les autres.

##### A. Les principes fondateurs du SCN

###### *Coopération locale, cohérence globale*

Au cœur du SCN, chaque liaison  $\omega_{i,j}$  se modifie selon une **décision locale** : deux entités évaluent la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  — qu'elle s'appuie sur une similarité, une co-information ou un gain de performance — puis ajustent  $\omega_{i,j}$ . Pourtant, cette dynamique locale affecte la **structure globale** : en synchronisant leurs mises à jour, des milliers ou des millions de connexions peuvent, en quelques itérations, donner naissance à des **macro-structures** ou des **clusters** (voir section 1.4.3).

###### *Adaptation continue*

Plutôt qu'un apprentissage ponctuel, le SCN procède par **itérations** successives. Les pondérations  $\omega_{i,j}$  se recalculent via une équation du type

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

où  $\eta$  indique le rythme de l'apprentissage et  $\tau$  limite la persistance d'un lien. Cette **dynamique** empêche toute stagnation et confronte en permanence l'évolution des connexions aux signaux de synergie qui apparaissent ou disparaissent.

###### *Mécanismes de régulation*

Pour qu'un trop grand nombre de liaisons ne se maintienne artificiellement, on peut **couper** celles dont la pondération demeure en deçà d'un certain **seuil**  $\omega_{\min}$ . À l'inverse, fixer un **plafond**  $\omega_{\max}$  empêche l'explosion de liens exagérément forts. Ces stratégies contribuent à la **parsimonie** du réseau : en n'y retenant que les connexions bénéfiques, on favorise la formation de sous-ensembles stables et la mise en évidence de motifs structurels plus riches.

##### B. Du réseau neuronal traditionnel au SCN

Un **réseau de neurones classique** se caractérise par une **architecture prédéfinie** : chaque couche contient un nombre fixe de neurones, et les connexions linéaires entre couches demeurent invariables. Bien que l'on ajuste les poids internes via la rétropropagation du gradient, la **topologie** — nombre de couches, disposition des neurones, schéma des connexions — reste, elle, **immuable**. À l'inverse, le **SCN** instaure un paradigme radicalement différent :

30. **La topologie évolue** spontanément. Les connexions peuvent apparaître ou disparaître en fonction de la **synergie** détectée.
31. Les entités elles-mêmes sont **actives** : elles disposent d'un état interne ou de mécanismes de représentation susceptibles d'évoluer pour améliorer leur **coopération**.
32. Le réseau peut **se réinventer** au gré de l'arrivée de nouvelles données ou de changements contextuels : on voit apparaître puis disparaître des **clusters** (micro-réseaux spécialisés), sans qu'un algorithme extérieur ne doive intervenir pour redéfinir la structure.

Avec cette **plasticité**, le SCN parvient à incorporer et à privilégier les synergies révélées par les données, adaptant la répartition des connexions bien plus librement qu'un réseau hiérarchique prédéterminé.

### C. Domaines d'application du SCN

33. **Multimodalité fluide**. En associant chaque modalité (vision, texte, audio, capteurs) à des entités spécialisées, le SCN crée et supprime des liaisons au gré de la **valeur ajoutée** qu'elles apportent. Les entités "visuelles" peuvent ainsi s'allier directement à des entités "textuelles" si elles trouvent un **gain mutuel**.
34. **Apprentissage continu et évolutif**. Dans un environnement changeant, le SCN **réajuste** sa configuration en continu, renforçant les liens utiles, délaissant les autres. Cette plasticité itérative apporte une **résilience** accrue, évitant tout réapprentissage intégral au moindre écart de distribution.
35. **Découverte de patrons "n-aires"**. Au-delà de l'interaction par paires, le SCN encourage la formation de **macro-clusters** quand plusieurs entités  $\{\mathcal{E}_1, \dots, \mathcal{E}_m\}$  coopèrent de manière à engendrer une plus-value collective supérieure à la somme de leurs synergies binaires.

### Conclusion

Le **Synergistic Connection Network (SCN)** illustre la **vision** fondamentale du DSL : un réseau **auto-organisé, évolutif**, où chaque entité se connecte, se réoriente ou se scinde suivant la **synergie** qu'elle détecte autour d'elle. À l'inverse d'un canevas hiérarchique traditionnel, cette organisation se déploie sans prescriptions initiales : elle se **bâtit** et se **rebâtit** au gré des liens les plus productifs, aboutissant à l'apparition de **clusters** et de **macro-structures** cohérentes, façonnées par l'apprentissage lui-même. Dans ce paradigme, l'interaction réciproque et la mise en commun de l'information prennent le pas sur la simple transmission d'un signal, créant une flexibilité particulièrement adaptée à la complexité et à la dynamique des données actuelles.

Dans la section suivante (1.4.3), nous verrons comment ces liens synergiques favorisent la naissance de **clusters auto-organisés**, et en quoi ceux-ci peuvent se regrouper en **macro-clusters** pour donner naissance à des entités d'information plus puissantes, ou plus abstraites, dans la démarche d'apprentissage distribué du Deep Synergy Learning.

#### 1.4.3. Notion de Cluster et de Macro-Cluster

Dans le **Synergistic Connection Network (SCN)**, décrit à la sous-section précédente (1.4.2), chaque entité d'information  $\mathcal{E}_i$  est reliée aux autres par des **pondérations synergiques**  $\omega_{i,j}(t)$  qui évoluent dans le temps. Ce mécanisme autorise l'**émergence spontanée** de sous-structures au sein du réseau : des groupes d'entités qui présentent entre elles une synergie élevée et tendent à **collaborer** plus fréquemment. On appelle couramment ces regroupements des **clusters**.

Au-delà de ces regroupements de base, il est possible que **plusieurs** clusters se regroupent encore à un niveau supérieur pour former des **macro-clusters**. Cette section (1.4.3) approfondit la *définition mathématique* de ces concepts, leurs *règles d'apparition* et leurs *conséquences* sur la dynamique d'apprentissage dans le **Deep Synergy Learning (DSL)**.

Dans la perspective du **Synergistic Connection Network (SCN)**, il est souvent utile de modéliser la structure courante du réseau par un **graphe**  $G(t)$  dont les **sommets**  $\{\mathcal{E}_1, \dots, \mathcal{E}_n\}$  correspondent aux entités d'information, et dont chaque **arête** est dotée d'une **pondération synergique**  $\omega_{i,j}(t)$ . À l'instant  $t$ , ces pondérations se réunissent dans une **matrice** :

$$W(t) = [\omega_{i,j}(t)]_{1 \leq i,j \leq n}.$$

Cette représentation matricielle capture simultanément l'ensemble des liens du réseau, chaque  $\omega_{i,j}(t)$  traduisant l'intensité de la coopération entre l'entité  $\mathcal{E}_i$  et l'entité  $\mathcal{E}_j$ . Dans un tel graphe, un **cluster**  $\mathcal{C} \subset \{1, \dots, n\}$  (c'est-à-dire un sous-ensemble des indices d'entités) se caractérise par des **connexions internes** plus **denses** ou plus **fortes** que ses connexions avec l'extérieur. Plusieurs manières existent pour formaliser cette notion de cluster :

Un **premier** point de vue consiste à **maximiser la somme** des pondérations internes. Plus précisément, on cherche un sous-ensemble  $\mathcal{C}$  maximisant



$$\sum_{i \in \mathcal{C}, j \in \mathcal{C}} \omega_{i,j}(t),$$

ce qui revient à rechercher la zone du graphe possédant la **densité** la plus élevée en liaisons fortes. Cette démarche peut être utile, par exemple, lorsque l'on souhaite mettre en évidence un noyau collaboratif de grande intensité.

Un **deuxième** critère, plus nuancé, repose sur le **rapport** entre la force interne et la force externe. On introduit alors une fonction de ratio

$$R(\mathcal{C}) = \frac{\sum_{i,j \in \mathcal{C}} \omega_{i,j}(t)}{\sum_{i \in \mathcal{C}, j \notin \mathcal{C}} \omega_{i,j}(t) + \epsilon},$$

où  $\epsilon > 0$  agit comme un **terme de régularisation** évitant les divisions par zéro. Le cluster  $\mathcal{C}$  optimal maximise alors  $R(\mathcal{C})$ , ce qui revient à privilégier des groupes dont la **cohésion interne** se révèle importante face aux connexions dirigées vers l'extérieur.

Enfin, un **troisième** point de vue fait appel à une **fonction d'énergie**  $\mathcal{J}$ , ou fonction de coût, qui **récompense** la densité intra-cluster tout en **pénalisant** les connexions externes (cf. section 1.4.2 pour une présentation plus générale du concept de coût dans le DSL). Les **clusters** apparaissent alors comme des **minima locaux** de  $\mathcal{J}$ . Cette perspective offre un cadre théorique aisément connectable à la physique statistique ou aux techniques d'optimisation combinatoire, permettant d'étudier la **stabilité** ou le **caractère** global/local des solutions.

Par-delà la variété de ces approches, le principe fondamental demeure : un **cluster** est un **groupe d'entités** qui, au sein de la matrice  $W(t)$ , entretient des liaisons relativement plus fortes entre ses membres qu'avec le reste du réseau. On retrouve ainsi l'idée intuitive d'une **collaboration** accrue à l'intérieur du sous-ensemble, associée à un **isolement** ou à une **différenciation** face aux entités extérieures. Dans la suite du texte, ces définitions de clusters s'avèrent essentielles pour analyser la manière dont le **SCN** laisse émerger automatiquement, au fil de la dynamique d'apprentissage, des **strates** coopératives plus ou moins vastes.

Dans le cadre du **Deep Synergy Learning**, les pondérations  $\omega_{i,j}(t)$  évoluent fréquemment selon la règle

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

où  $\eta > 0$  et  $\tau > 0$  sont des constantes, tandis que  $S(\mathcal{E}_i, \mathcal{E}_j)$  désigne la **synergie** entre les entités  $i$  et  $j$ . Dès lors qu'une pondération  $\omega_{i,j}(t)$  dépasse un seuil  $\omega_{\min}$ , on considère qu'il existe un **lien effectif** entre les deux entités. Plusieurs phénomènes concourent alors à la **formation d'un cluster** :

- **Renforcement interne.** Lorsqu'un certain **groupe**  $\mathcal{C} \subset \{1, \dots, n\}$  présente des synergies élevées entre ses membres, les pondérations  $\omega_{i,j}(t)$  associées ont tendance à **augmenter** pour tous  $i, j \in \mathcal{C}$ .
- **Isolement progressif.** À mesure que les liens internes se consolident, les connexions externes (i.e. celles dont la synergie avec les entités extérieures demeure plus faible) s'**atténuent**, rendant le sous-ensemble  $\mathcal{C}$  de plus en plus autonome et homogène.

Ce processus engendre un **cluster** cohérent, formé sans qu'aucun agencement préalable ne soit imposé.

#### Exemple de scénario d'émergence

36. **Temps initial (t=0).** Les pondérations  $\omega_{i,j}(0)$  sont faibles ou distribuées de manière uniforme, si bien qu'aucune structure évidente ne se détache.
37. **Interactions locales.** Certaines entités détectent une synergie ( $S(\mathcal{E}_i, \mathcal{E}_j) > 0$ ) suffisamment avantageuse pour se renforcer localement. Les  $\omega_{i,j}(t)$  correspondantes s'amplifient.
38. **Formation d'une "graine".** Progressivement, un **mini-cluster** naît sous la forme de 2 ou 3 entités dont les liens internes, devenus plus élevés, assurent un noyau de haute synergie.
39. **Attraction.** Si une entité  $\mathcal{E}_k$  à la périphérie présente aussi une bonne synergie avec cette graine, ses liaisons  $\omega_{k,i}$  ( $i \in \mathcal{C}$ ) se renforcent, ce qui intègre  $\mathcal{E}_k$  au cluster en expansion.
40. **Consolidation.** Une fois que les **liens internes** atteignent un niveau stable et que les connexions vers l'extérieur restent faibles, le groupe  $\mathcal{C}$  se **stabilise** comme cluster. D'un point de vue formel, cela correspond à un **minimum local** d'une fonction d'énergie  $\mathcal{J}$ , où  $\mathcal{C}$  se comporte comme un **attracteur**.

Au-delà des **clusters** de base, il n'est pas rare que plusieurs sous-groupes finis se **rassemblent** pour former un **macro-cluster**, c'est-à-dire une entité supérieure englobant plusieurs clusters déjà constitués.

Pour cela, on considère un **méta-groupe**  $\mathcal{M}$  rassemblant des **clusters**  $\mathcal{C}_1, \mathcal{C}_2, \dots$ . Les pondérations entre deux clusters  $\mathcal{C}_1$  et  $\mathcal{C}_2$  se définissent par

$$\Omega(\mathcal{C}_1, \mathcal{C}_2) = \frac{1}{|\mathcal{C}_1| \cdot |\mathcal{C}_2|} \sum_{\substack{i \in \mathcal{C}_1 \\ j \in \mathcal{C}_2}} \omega_{i,j}(t).$$

Si  $\Omega(\mathcal{C}_1, \mathcal{C}_2)$  dépasse un certain niveau, les deux groupes **fusionnent**, produisant ainsi un **macro-cluster** élargi.

#### Processus de fusion

41. **Clusters initiaux.** Supposons que  $\mathcal{C}_1$  et  $\mathcal{C}_2$  soient déjà établis.
42. **Renforcement inter-clusters.** Dès lors que plusieurs liaisons  $\omega_{i,j}$  ( $i \in \mathcal{C}_1, j \in \mathcal{C}_2$ ) se consolident, la synergie globale entre  $\mathcal{C}_1$  et  $\mathcal{C}_2$  augmente.
43. **Fusion structurelle.** Sitôt ce renforcement assez prononcé pour franchir un **seuil**, les deux clusters cessent de constituer des sous-groupes autonomes : ils forment un **macro-cluster**  $\mathcal{M}$ .
44. **Vue hiérarchique.** Cette démarche rappelle le **clustering hiérarchique agglomératif**, à la différence majeure que, dans le DSL, les pondérations  $\omega_{i,j}$  ne sont pas statiques mais se **règlent** en permanence, autorisant une consolidation plus dynamique.

Pour formaliser ces phénomènes de clusterisation, on introduit couramment une **fonction d'énergie**  $\mathcal{J}$  :

$$\mathcal{J}(\{\omega_{i,j}\}) = - \sum_{(i,j)} \omega_{i,j} S(\mathcal{E}_i, \mathcal{E}_j) + \alpha \sum_{(i,j)} (\omega_{i,j})^2 + \dots$$

- Le premier terme **récompense** les liaisons de haute synergie : chercher à **minimiser**  $\mathcal{J}$  revient donc à **maximiser** la somme  $\sum \omega_{i,j} S(\mathcal{E}_i, \mathcal{E}_j)$ .
- Le second terme (avec  $\alpha > 0$ ) introduit une forme de **régularisation** ou de “penalty” pour empêcher une croissance excessive et simultanée de tous les liens.

Dans ce cadre, un **cluster** renvoie à un **minimum local** de  $\mathcal{J}$ , où l'on observe un niveau élevé de pondérations **en interne**. Un **macro-cluster** surgit quand on identifie un autre minimum local, rassemblant plusieurs groupes préexistants.

L'émergence de **clusters** segmente le Synergistic Connection Network en **régions** spécialisées (p. ex. un cluster associant vision et texte, un autre combinant audio et capteurs, etc.). On obtient ainsi une forme de **division du travail** au sein d'un réseau de grande taille, ce qui favorise la **lisibilité** et la **résilience** globale.

Par ailleurs, la formation de **macro-clusters** peut être rapprochée de l'idée de **modules cognitifs**, dans lesquels plusieurs sous-ensembles d'entités se lient pour s'attaquer à des tâches plus complexes. Le réseau acquiert de la sorte une **modularité adaptative**, qu'il s'agisse de manipuler différents canaux de données ou d'élaborer des représentations de plus haut niveau.

D'un point de vue **algorithmique**, détecter et gérer explicitement les clusters permet de segmenter la **mise à jour** des pondérations : plutôt que d'analyser un vaste graphe dans sa globalité, on peut restreindre certains calculs à des modules faiblement connectés entre eux. Cette **auto-organisation** en groupes soudés se démarque d'un simple “clustering statique”, car les pondérations  $\omega_{i,j}(t)$  évoluent continuellement, autorisant des scissions, des fusions et même la disparition de clusters entiers au gré des fluctuations de la synergie.

#### Exemples d'application

- **Analyse multimodale en temps réel.** Dans une application de surveillance, on peut regrouper en un même cluster les entités traitant l'image et la parole (cluster “visage + discours”), et dans un autre, celles dédiées à la détection de mouvements anormaux ou de sons inhabituels. Les agrégats évoluent de manière dynamique selon la synergie perçue entre différents flux (lumière, son, déplacements, etc.).
- **Recommandation et filtrage collaboratif.** En représentant les utilisateurs et les contenus par des entités, les connexions  $\omega_{u,c}$  sont élevées quand un utilisateur  $u$  apprécie un contenu  $c$ . La dynamique du réseau provoque l'émergence de **clusters** d'utilisateurs ayant des goûts proches et de contenus similaires ; un **macro-cluster** peut alors regrouper un large groupe d'utilisateurs et l'ensemble des contenus qui leur plaisent collectivement.
- **Traitement biologique ou neuroscientifique.** Si l'on assimile les neurones (ou de petites populations neuronales) à des entités  $\mathcal{E}_i$ , on peut observer la formation de **clusters** assimilables à des assemblées neuronales locales, puis la mise en place de **macro-clusters** associant plusieurs aires cérébrales interconnectées. Ceci traduit une **organisation fonctionnelle** plus étendue de la dynamique neuronale.

## Conclusion

La **formation de clusters** (puis de **macro-clusters**) incarne un des atouts majeurs du **DSL** : le réseau peut **s'auto-structurer** en sous-ensembles flexibles, chacune de ces composantes se stabilisant ou fusionnant selon la synergie qui émerge dans le temps. Surtout, il ne s'agit pas d'une simple classification unique et immuable : les pondérations  $\omega_{i,j}(t)$  continuent de se réévaluer, donnant lieu à des réorganisations spontanées.

Dans la suite (sections 1.4.4 à 1.4.7), nous examinerons les bases mathématiques qui sous-tendent ce phénomène, en décrivant la **fonction de synergie** (distance, similarité ou information mutuelle), la façon dont les pondérations  $\omega_{i,j}$  s'adaptent dans le temps, la distinction entre **interactions directes** et **indirectes**, puis la question de la **synergie n-aire**, où plusieurs entités coopèrent simultanément au-delà des simples relations binaires.

### 1.4.4. Fonctions de Synergie : Distance, Similarité et Co-Information

Au cœur du **Deep Synergy Learning (DSL)**, les *pondérations synergiques*  $\omega_{i,j}$  reliant deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  évoluent en fonction d'une **mesure de synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$ . Cette fonction  $S$  évalue dans quelle mesure la **coopération** entre deux entités apporte un **gain** supérieur à l'utilisation isolée de leurs informations.

Bien que le **principe** de la synergie demeure le même (capturer l'apport mutuel), il existe **plusieurs manières** de la **définir** et de la **calculer** dans la pratique. Les plus communes reposent sur (1) des **distances** entre représentations, (2) des **similarités** (souvent normalisées), ou (3) des mesures d'**information** (entropie, co-information, etc.). Cette section expose en détail ces approches, avec une formulation mathématique approfondie.

Dans le **Deep Synergy Learning (DSL)**, la pondération  $\omega_{i,j}(t)$  reliant deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  suit fréquemment une équation de la forme

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

où  $\eta$  et  $\tau$  sont des hyperparamètres positifs, et  $S(\mathcal{E}_i, \mathcal{E}_j)$  la **fonction de synergie**. Cette fonction reflète dans quelle mesure les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  estiment pouvoir coopérer avantageusement. Dès lors,

- Si  $S(\mathcal{E}_i, \mathcal{E}_j)$  est **positif** et assez grand, la liaison  $\omega_{i,j}$  se **renforce**.
- Si  $S(\mathcal{E}_i, \mathcal{E}_j)$  est faible ou négatif, la liaison s'**affaiblit**.

Le **choix** de la fonction  $S$  influe directement sur la **topologie** finale du réseau : deux entités jugées proches ou complémentaires se retrouveront dans un même **cluster** (voir section 1.4.3), tandis que des entités peu compatibles verront leur lien s'étioler. Différentes approches existent pour définir cette synergie.

#### 1. Synergie fondée sur la distance

Une façon intuitive d'instaurer la synergie est de partir d'une **distance**  $d(\mathcal{E}_i, \mathcal{E}_j)$  calculée sur leurs représentations, souvent des vecteurs  $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^d$ . Un exemple répandu consiste en la **distance euclidienne**  $\|\mathbf{x}_i - \mathbf{x}_j\|$ . On convertit ensuite cette distance en un score de synergie **décroissant** :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \frac{1}{1 + \|\mathbf{x}_i - \mathbf{x}_j\|^2},$$

ou bien, en autorisant des valeurs négatives, on peut poser

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

L'idée est qu'une faible distance se traduit par une **forte synergie**, tandis qu'une distance importante produit un score négatif ou tendant vers zéro. Il est fréquent d'ajouter un **paramètre d'échelle**  $\sigma > 0$  pour réguler la sensibilité aux écarts, par exemple

$$S(\mathcal{E}_i, \mathcal{E}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2).$$

Cette fonction valorise la synergie pour les entités **très proches** et la fait décroître rapidement au-delà d'un rayon  $\sigma$ . La distance choisie dépend ensuite de la nature des données : L1, L2, distance sur des tenseurs d'images ou spectrogrammes, etc.

#### 2. Synergie fondée sur la similarité

Un autre choix repose sur une **similarité** plutôt que sur une distance. Par exemple, la **similarité cosinus** :

$$\text{sim}(\mathbf{x}_i, \mathbf{x}_j) = \frac{\langle \mathbf{x}_i, \mathbf{x}_j \rangle}{\|\mathbf{x}_i\| \|\mathbf{x}_j\|},$$

qui varie dans  $[-1, 1]$ . On peut alors définir

$$S(\mathcal{E}_i, \mathcal{E}_j) = \max(0, \text{sim}(\mathbf{x}_i, \mathbf{x}_j)) \quad \text{ou} \quad S(\mathcal{E}_i, \mathcal{E}_j) = 1/2 (\text{sim}(\mathbf{x}_i, \mathbf{x}_j) + 1),$$

histoire de normaliser la valeur dans  $[0, 1]$ . Cette approche est pratique quand on souhaite regrouper des **vecteurs** se ressemblant en direction ou quand on manipule des **corrélations** (ex. coefficient de Pearson). Elle est parfois moins adaptée si l'on cherche à capter la complémentarité non linéaire entre des entités fortement dissemblables.

### 3. Synergie fondée sur la co-information ou l'information mutuelle

Lorsque les entités  $\mathcal{E}_i, \mathcal{E}_j$  renvoient à des **variables aléatoires**  $\mathbf{X}_i, \mathbf{X}_j$ , on peut évaluer la synergie via des **quantités entropiques** :

$$I(\mathbf{X}_i; \mathbf{X}_j) = H(\mathbf{X}_i) + H(\mathbf{X}_j) - H(\mathbf{X}_i, \mathbf{X}_j),$$

qui mesure l'**information mutuelle**. Toutefois, cette mesure ne différencie pas la simple **redondance** d'une véritable **coopération**. On peut alors recourir à la **co-information** ou à la **Partial Information Decomposition (PID)**, laquelle décompose l'information partagée en portions "synergique" et "redondante". Cela donne un score :

$$S(\mathcal{E}_i, \mathcal{E}_j) = \max(0, I_{\text{syn}}(\mathbf{X}_i, \mathbf{X}_j | \mathbf{Y})),$$

où  $I_{\text{syn}}$  désigne la contribution strictement synergique, au sens de la définition employée (co-info, PID, etc.). Cette méthode peut détecter des interactions non linéaires et complexes, mais elle se révèle plus **onéreuse** à calculer, en particulier pour des données de haute dimension, et nécessite de **choisir** la mesure d'information la plus pertinente au contexte.

### Conseils pour le choix de la fonction $S$

Le **DSL** ne prescrit pas une forme unique de synergie, mais un **cadre** flexible où n'importe quelle mesure d'**apport mutuel** entre entités peut convenir. On retient néanmoins certains principes :

45. **Distance.** Simple à implémenter et intuitive, elle va favoriser l'association d'entités similaires (au sens d'un espace de représentation). Elle sous-estime parfois la complémentarité de deux entités très différentes mais hautement coopératives.
46. **Similarité.** Plus adaptée dès lors qu'on compare des directions ou des vecteurs déjà normalisés. Les grandes similarités indiquent une forte parenté, mais la complémentarité non linéaire peut lui échapper.
47. **Information mutuelle / co-info / PID.** Très puissante pour les **coopérations** complexes et non linéaires, mais coûteuse en ressources de calcul et sensible aux méthodes d'estimation statistique.

Il est évidemment possible de **combiner** plusieurs indicateurs (distance, similarité, information mutuelle) ou de pondérer différentes composantes pour confectionner un score  $S(\cdot, \cdot)$  plus adapté. Le choix exact dépendra de la nature des données, du domaine d'application et des **objectifs** poursuivis (clustering, prédiction, fusion multimodale, etc.).

### Exemples d'utilisation

48. **Analyse d'images.** Considérons des entités  $\mathcal{E}_i$  représentant des patchs ou des régions d'images. Une distance L2 sur des vecteurs de caractéristiques (par ex. issus d'un réseau de neurones convolutionnel) peut fournir un score de synergie décroissant : plus les patchs se ressemblent, plus ils coopèrent.
49. **Fusion multimodale.** Supposons deux entités  $\mathbf{X}_{\text{audio}}$  et  $\mathbf{X}_{\text{visuel}}$ . On mesure leur co-information conditionnellement à une variable  $\mathbf{Y}$  (classe d'événement). Si leur combinaison apporte un gain clair, on augmente la synergie et, par conséquent, on renforce leur lien.
50. **Système hybride symbolique-connexionniste.** Un module symbolique (décrivant des règles ou faits abstraits) et un module vectoriel (des embeddings d'images, par exemple) peuvent relier leurs entités si l'information mutuelle ou la similarité cosinus démontre qu'ils s'améliorent réciproquement.

### Conclusion et ouverture

La fonction de synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$  est la **pièce angulaire** du Deep Synergy Learning. Selon qu'on s'appuie sur une **distance**, une **similarité** ou une **information mutuelle**, on oriente la manière dont les entités **coopèrent** et se **relient** au sein du **Synergistic Connection Network**.

- Les **fonctions de distance** favorisent l'agrégation d'éléments proches dans l'espace des caractéristiques.

- Les **fonctions de similarité** rassemblent des vecteurs corrélés, tout en ignorant parfois la complémentarité.
- Les **mesures entropiques** (co-info, PID, etc.) explorent la coopération profonde, prenant en compte les facettes non linéaires et la contribution uniquement révélée par l'association des entités.

Les sections suivantes (1.4.5 à 1.4.7) approfondiront la **mise à jour temporelle** de  $\omega_{i,j}(t)$  fondée sur cette synergie, la distinction entre **interactions directes** et **indirectes**, et la manière de gérer la **synergie n-aire**. L'ensemble de ces points complète la **vision** du DSL comme un réseau évolutif, autonome et apte à faire émerger des **structures** d'apprentissage complexe dans des contextes variés.

#### 1.4.5. Pondérations Adaptatives et Évolution Temporelle

Les sections précédentes (1.4.1 à 1.4.4) ont établi les **fondements** de l'architecture générale du **Deep Synergy Learning (DSL)** : chaque **entité** peut se relier à d'autres par des **pondérations synergiques**  $\omega_{i,j}$  qui reflètent la **valeur ajoutée** d'une coopération. Reste à décrire **comment** ces pondérations **changent** dans le temps, c'est-à-dire la **loi d'évolution** qui fait que le réseau se **reconfigure** en permanence. C'est l'objet de cette sous-section (1.4.5), qui explique :

- Les **équations** gouvernant l'adaptation des pondérations,
- Le sens de cette **dynamique** (renforcement / affaiblissement),
- Les implications sur la **convergence** ou la **stabilité** des liens,
- Le rôle des **paramètres** (taux d'apprentissage, régularisation, etc.) dans l'auto-organisation du réseau.

Cette **évolution temporelle** est au cœur du **SCN** (Synergistic Connection Network), car c'est elle qui autorise la création progressive de **clusters** et, par extension, la modélisation d'un **apprentissage** réellement continu.

Dans la plupart des formulations du **Deep Synergy Learning (DSL)**, la pondération  $\omega_{i,j}(t)$  reliant deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  se met à jour suivant un schéma adaptatif de la forme

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

où  $\eta$  désigne un **taux d'apprentissage** (ou **pas de mise à jour**) et  $\tau$  un **coefficient de régulation** (ou *terme d'oubli*). La fonction  $S(\mathcal{E}_i, \mathcal{E}_j)$  correspond à la mesure de **synergie** choisie (distance, similarité, co-information, etc.). Ce cadre de mise à jour s'interprète ainsi :

- Lorsque la synergie  $S$  entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$  est assez élevée, et que  $\omega_{i,j}(t)$  reste encore modérée, le terme  $\eta [S - \tau \omega]$  demeure **positif**, accroissant la liaison à chaque itération.
- À l'inverse, si la synergie est faible (ou négative), ou si  $\omega_{i,j}(t)$  est déjà trop grande, le terme devient **négatif**, ce qui affaiblit la liaison.

On peut mettre en évidence un **point fixe** en imposant  $\omega_{i,j}(t+1) = \omega_{i,j}(t) = \omega_{i,j}^*$ . La condition associée,

$$\omega_{i,j}^* = S/\tau,$$

montre que plus la **synergie** est grande, plus la **pondération d'équilibre**  $\omega_{i,j}^*$  atteint un niveau élevé.

#### Variantes et enrichissements

Plusieurs adaptations sont possibles autour de cette règle. On peut retenir un **historique** afin de lisser la dynamique ou pondérer les valeurs antérieures  $\omega_{i,j}(t-1)$ . La forme de mise à jour elle-même peut être **non linéaire** : on quitte la version linéaire  $\omega + \eta [S - \tau \omega]$  pour des équations saturantes, exponentielles ou inspirées des règles "hebbiennes". De plus, la synergie  $S$  peut varier avec le temps si les entités  $\mathcal{E}_i(t)$  et  $\mathcal{E}_j(t)$  mettent à jour leurs représentations internes ; la pondération  $\omega_{i,j}(t)$  influence alors la synergie, qui en retour modifie la structure globale.

- pour les signes de qualité relevant de l'audit énergétique, un bilan des audits présentés comme référence ;

- pour les signes de qualité relevant de l'installation d'unités de production d'électricité photovoltaïque, un bilan des audits présentés comme référence ;

- le nombre d'audits effectués et une synthèse générale des résultats et écarts constatés et des sanctions appliquées à l'issue des contrôles ;

## Système dynamique discret ou continu

Le modèle

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

est une **équation aux différences**, interprétable comme un **système dynamique** discret dans un espace de dimension  $\binom{n}{2}$  (ou  $n(n-1)$  selon que l'on distingue la direction  $i \rightarrow j$ ). On y analyse notamment :

51. Les **points fixes**  $\omega_{i,j}^*$  satisfaisant

$$\omega_{i,j}^* = \omega_{i,j}^* + \eta [S - \tau \omega_{i,j}^*].$$

52. La **stabilité** de ces points. Par exemple, en linéarisant autour de l'équilibre, on obtient

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

montrant que la condition  $\eta \tau < 1$  est usuellement requise pour maintenir une évolution stable des pondérations.

L'équivalent **en temps continu** est donné par l'ODE

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

Ici encore, la solution tend vers  $\omega_{i,j}^* = S/\tau$  s'il ne survient pas d'autres perturbations ni dépendances supplémentaires.

### Seuils, cap et parcimonie

Dans le **DSL**, on introduit souvent un **seuil**  $\omega_{\min}$  en deçà duquel la connexion est jugée inexistante :

$$\omega_{i,j}(t) < \omega_{\min} \Rightarrow \text{aucun lien effectif entre } i \text{ et } j.$$

Même si l'équilibre  $\omega_{i,j}^*$  est légèrement positif, la liaison doit dépasser  $\omega_{\min}$  pour être considérée. À l'autre extrémité, on peut imposer un **cap**  $\omega_{i,j}(t) \leq \omega_{\max}$  pour freiner l'augmentation des pondérations et éviter les valeurs extrêmes. Sur un **diagramme de phase** ( $\omega-\dot{\omega}$ ), ces deux seuils garantissent une **parcimonie** structurale : seuls les liens ayant démontré une synergie conséquente survivent, tandis que ceux qui resteraient trop faibles ou monteraient exagérément se trouvent coupés ou plafonnés.

### Interdépendance globale

Dans un réseau comptant  $n$  entités, la synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$  peut dépendre d'autres entités (cf. section 1.4.4). On écrit parfois

$$S(\mathcal{E}_i, \mathcal{E}_j \mid \{\mathcal{E}_k\}_{k \neq i,j}),$$

pour souligner le fait qu'une **co-information** conditionnelle peut influencer sur la pondération. Les **misés à jour** des  $\omega_{i,j}$  sont alors **couplées** : la liaison  $i \leftrightarrow j$  se renforce ou s'affaiblit en interaction avec d'autres liaisons  $j \leftrightarrow k$ , etc. C'est ce couplage qui fait émerger des **clusters** (section 1.4.3) : les entités formant un sous-groupe  $\mathcal{C}$  stabilisent mutuellement leurs pondérations internes, aboutissant à une "bulle" de coopération élevée, tandis que les liens extérieurs, moins profitables, s'atténuent.

### Feedback et plasticité

Interpréter la dynamique du DSL comme un système à **feedback** offre un éclairage pertinent :

53. **Feedback positif.** Dès lors qu'une liaison  $\omega_{i,j}$  grandit (parce que  $S > 0$ ),  $\mathcal{E}_i$  et  $\mathcal{E}_j$  accroissent encore leur coopération, consolidant le lien.
54. **Feedback négatif.** Si, au contraire, la liaison s'emballe, le terme  $-\tau \omega_{i,j}(t)$  ou une autre pénalisation (par ex. un terme cubique) freine cette croissance, garantissant la **stabilisation**.

De façon analogique, on peut s'inspirer de la **plasticité synaptique** :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta (a_i(t) a_j(t)) - \alpha \omega_{i,j}(t),$$

où  $a_i(t)$  et  $a_j(t)$  sont les “activités” ou “états” des entités, vus comme un substitut instantané à la synergie  $S_{i,j}(t)$ . Par ailleurs, l’usage d’un **terme cubique**  $-\beta (\omega_{i,j})^3$  agit comme une stabilisation encore plus marquée, évitant la croissance sans borne.

#### Impacts et conclusions

55. **Flexibilité continue.** Grâce à l’actualisation en continu de  $\omega_{i,j}(t)$ , le **Synergistic Connection Network (SCN)** s’adapte sans cesse aux nouveaux flux ou aux modifications d’environnement, bien plus qu’un réseau à architecture fixe.
56. **Formation de clusters.** Les entités liées par une synergie forte consolident leurs liaisons, s’assemblent en **clusters**, et peuvent se **fusionner** en macro-clusters (voir section 1.4.3). Cette recomposition est fluide et dépend directement de la dynamique des pondérations.
57. **Régime stable vs. oscillatoire.** Selon les choix de synergie, de  $\eta$ , de  $\tau$ , etc., le réseau peut tendre vers un état stable, osciller, ou coexister dans plusieurs configurations d’équilibre.
58. **Aspects computationnels.** Mettre à jour toutes les paires  $(i,j)$  exige un coût a priori en  $O(n^2)$ . Des heuristiques (sparse updates, random sampling) peuvent s’avérer nécessaires pour gérer de grands  $n$  de façon **scalable**.

Cette **dynamique adaptative** des pondérations constitue le moteur de l’**auto-organisation** dans le DSL, permettant de découvrir et de consolider des **sous-structures** au sein du réseau. Les sections suivantes (1.4.6 et 1.4.7) s’intéresseront à la **distinction entre interactions directes et indirectes** et à la **synergie n-aire**, deux aspects cruciaux pour comprendre la **richesse** et la **réactivité** de ce paradigme d’apprentissage.

#### 1.4.6. Interactions Directes et Indirectes

Jusqu’à présent, la description du **Deep Synergy Learning (DSL)** s’est essentiellement focalisée sur la **relation directe** entre deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ , mesurée par une **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  et traduite dans une **pondération**  $\omega_{i,j}$ . Pourtant, dans un réseau où de nombreuses entités coexistent, les **interactions indirectes** jouent un rôle crucial : deux entités qui n’ont pas de liaison directe peuvent tout de même s’influencer si elles sont **connectées** via une **chaîne** (ou un **chemin**) de plusieurs liens synergiques. Cette section (1.4.6) explique :

59. Comment les **interactions** peuvent se propager à travers le **Synergistic Connection Network (SCN)**,
60. Comment la **synergie** entre deux entités peut être **modulée** par leurs relations avec d’autres,
61. Quelles implications mathématiques découlent de ces boucles d’influence plus complexes, notamment pour la **formation** de clusters et l’**émergence** de comportements globaux.

Cette problématique est centrale dans la compréhension de l’**auto-organisation** : même si chaque entité ne considère que les liens qui la concernent directement, l’ensemble du réseau peut manifester des effets d’**influence à distance** ou de **coordination** en chaîne.

On appelle **interaction indirecte** entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$  toute séquence  $\mathcal{E}_i \rightarrow \mathcal{E}_k \rightarrow \dots \rightarrow \mathcal{E}_m \rightarrow \mathcal{E}_j$  dans laquelle chacune des paires  $(\mathcal{E}_u, \mathcal{E}_v)$  du chemin possède une pondération  $\omega_{u,v}(t)$  significative. Le **Synergistic Connection Network (SCN)** autorise ainsi la propagation d’**influence** ou de **coopération** le long d’un **chemin** reliant deux entités, même si ces dernières ne sont pas directement connectées.

#### 1. Principes généraux : chemins et relai d’information

Lorsque la liaison  $\omega_{k,m}$  est élevée, l’entité  $\mathcal{E}_k$  peut **transmettre** (ou “relayer”) certaines données ou incitations à  $\mathcal{E}_m$ . C’est ainsi qu’une synergie faible ou nulle entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$  peut progressivement **s’intensifier** si elles interagissent via des entités communes  $\{\mathcal{E}_k, \mathcal{E}_\ell, \dots\}$  qui procurent un **gain mutuel**. De même, si l’on modélise la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  comme une quantité **contextuelle**, on peut définir

$$S(\mathcal{E}_i, \mathcal{E}_j \mid \mathcal{E}_k),$$

qui tient compte du fait que la coopération entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$  peut être **facilitée** ou **rendue moins nécessaire** par la présence d’une entité pivot  $\mathcal{E}_k$ .

#### Exemple : pivot et redondance

- Si  $\mathcal{E}_k$  remplit déjà les rôles combinés de  $\mathcal{E}_i$  et  $\mathcal{E}_j$ , l’intérêt pour  $\mathcal{E}_i$  et  $\mathcal{E}_j$  de se lier **directement** diminue (redondance).
- Au contraire, si  $\mathcal{E}_k$  fournit un complément utile à  $\mathcal{E}_i$  et  $\mathcal{E}_j$ , ce **contexte** peut augmenter leur synergie bilatérale, incitant l’émergence d’un lien direct  $\omega_{i,j}$ .

## 2. Notion de chemins et puissance de la matrice $W$

Le **graphe** du SCN se décrit par la matrice pondérée  $W(t)$ , de taille  $n \times n$ , dont l'entrée  $\omega_{i,j}(t)$  indique la connexion directe entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . Lorsqu'on considère un **chemin** de longueur 2 entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$  via un intermédiaire  $\mathcal{E}_k$ , la multiplication de matrices révèle que  $(W^2)_{i,j} = \sum_{k=1}^n \omega_{i,k}(t) \omega_{k,j}(t)$ . De manière analogue,  $(W^p)_{i,j}$  réunit l'effet de **tous** les chemins de longueur  $p$  liant  $i$  à  $j$ .

Si la dynamique des pondérations reste **relativement stable**, l'examen des puissances  $W^2, W^3, \dots$  peut dévoiler des **interactions indirectes** (chemins multiples ou plus longs). Des valeurs élevées de  $(W^p)_{i,j}$  traduisent une **forte influence** de  $\mathcal{E}_i$  sur  $\mathcal{E}_j$  à travers plusieurs intermédiaires. En présence de **cycles** (ex.  $\mathcal{E}_i \rightarrow \mathcal{E}_k \rightarrow \mathcal{E}_m \rightarrow \mathcal{E}_i$ ), on peut observer des effets d'**amplification** : si chaque liaison du cycle se renforce, cela accroît la synergie circulant en boucle.

D'un point de vue **linéaire**, la **valeur propre** de plus grande amplitude (rayon spectral  $\rho(W)$ ) reflète la **stabilité** du système :

- Si  $\rho(W)$  excède 1, de petites perturbations internes risquent de se propager et de s'amplifier (effet possible d'oscillation).
- Si  $\rho(W) < 1$  (après régulation), la dynamique a tendance à converger vers un état stable.

## 3. Contexte et synergie conditionnelle

Dans un réseau synergique, l'existence de **chemins indirects** modifie considérablement la manière dont deux entités  $\mathcal{E}_i, \mathcal{E}_j$  perçoivent leur **coopération**. Même si  $\omega_{i,j}$  démarre faible, il peut se consolider via l'influence d'entités tierces  $\mathcal{E}_k$ . Formellement, on modélise cette dépendance par une fonction

$$S(\mathcal{E}_i, \mathcal{E}_j | \{\omega_{i,k}, \omega_{k,j}, \dots\}),$$

introduisant un **système dynamique non linéaire**. Les équations des pondérations deviennent

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

La dépendance entre les  $\omega_{i,j}$  fait naître des **boucles de feedback**, où un renforcement dans un segment du réseau rejaillit sur d'autres liaisons, et ainsi de suite. En pratique, l'**analyse** de stabilité ou de convergence peut impliquer des techniques avancées (théorie des bifurcations, méthodes de Lyapunov, etc.).

## 4. Rôle des interactions indirectes dans la dynamique globale

Les **interactions indirectes** expliquent comment un réseau DSL parvient à :

- **Faire émerger** de nouveaux liens : si  $\mathcal{E}_i$  et  $\mathcal{E}_j$  constatent qu'elles coopèrent déjà via un pivot  $\mathcal{E}_k$ , elles peuvent juger bon de forger une connexion directe  $\omega_{i,j} > 0$ .
- **Fusionner** des clusters : deux communautés initialement distinctes, reliées par quelques chemins transitifs, peuvent se reconnaître mutuellement profitables et **coalescer** en un macro-cluster.
- **Gérer** la circulation de bruit : une entité bruyante ou imprécise peut nuire indirectement à des voisins qui ne sont pourtant pas liés à elle de façon directe, tandis qu'un intermédiaire de confiance peut rehausser la fiabilité des échanges.

Ce mécanisme d'**influence à distance** se retrouve dans des domaines variés. En **robotique multi-agents**, un agent  $\mathcal{E}_i$  peut influencer un autre  $\mathcal{E}_j$  via un agent  $\mathcal{E}_k$  jouant le rôle de relai ; la structure d'échange se reconfigure selon la performance observée. En **neurosciences**, de nouvelles synapses directes se forment après une longue période où l'activité neuronale transite par plusieurs synapses intermédiaires.

## Conclusion et ouverture

Les **interactions indirectes** forment un pivot crucial pour saisir l'**auto-organisation** d'un Synergistic Connection Network (SCN). Deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  initialement faiblement connectées peuvent se **rapprocher** grâce aux chemins multiples les reliant via des tiers, tout



comme des boucles internes peuvent amplifier ou réguler la synergie. Cette dimension de **propagation** et de **contexte** est essentielle pour comprendre la **formation** de macro-structures, la dynamique des **cycles** et la **coévolution** des pondérations dans le DSL.

La section suivante (1.4.7) examinera de plus près la **synergie binaire** et surtout la **synergie n-aire**, c'est-à-dire l'impact de coopérations impliquant simultanément plus de deux entités, un aspect qui renforce encore la **richesse** des comportements émergents dans le Deep Synergy Learning.

#### 1.4.7. Synergie binaire et n-aire : au-delà des relations deux à deux

La **synergie binaire** demeure la forme la plus couramment utilisée pour évaluer le degré de coopération entre deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . On définit pour cela une fonction  $S(\mathcal{E}_i, \mathcal{E}_j)$  (pouvant être fondée sur une distance, une similarité, de la co-information, etc.) qui mesure dans quelle mesure

$$\mathbf{x}_i (+ \mathbf{s}_i, \theta_i) \Rightarrow \mathbf{x}_j (+ \mathbf{s}_j, \theta_j)$$

s'enrichissent mutuellement ou, au contraire, s'avèrent redondantes ou peu utiles l'une à l'autre.

- **Mise à jour.** À chaque itération, la pondération  $\omega_{i,j}(t)$  se voit recalculée en fonction de la **valeur**  $S(\mathcal{E}_i, \mathcal{E}_j)$ .
- **Clusters.** Les clusters décrits en section 1.4.3 émergent généralement d'une **somme** ou d'une **agrégation** de ces liaisons binaires (souvent assortis d'un seuil, stabilisant la formation de groupes plus étendus).

Toutefois, la synergie binaire suppose que l'apport mutuel entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$  ne dépende **pas** explicitement des autres entités. Or, la section 1.4.6 a montré que le **contexte** global (présence de  $\mathcal{E}_k, \dots$ ) peut influencer de manière significative : certaines lacunes peuvent être comblées, certaines informations complémentaires se mutualiser, ce qui peut révéler une **coopération** émergente entre plusieurs entités à la fois.

#### 1. Synergie n-aire : idée d'un effet collectif

La **synergie n-aire** engage un **ensemble**  $\{\mathcal{E}_{k_1}, \dots, \mathcal{E}_{k_m}\}$  (avec  $m \geq 3$  dans la plupart des cas), dont la **combinaison** fournit une valeur ajoutée qu'on ne saurait réduire à la somme des synergies binaires. On écrit

$$S_n(\mathcal{E}_{k_1}, \dots, \mathcal{E}_{k_m})$$

pour désigner la synergie qui s'exprime **simultanément** entre ces  $m$  entités. On veut y saisir l'idée que "toutes ensemble, elles valent davantage que la somme de leurs collaborations deux à deux". Pour le cas de trois entités  $\mathcal{E}_a, \mathcal{E}_b, \mathcal{E}_c$ , on parle parfois de **complémentarité stricte** :

$$S_n(\mathcal{E}_a, \mathcal{E}_b, \mathcal{E}_c) > S(\mathcal{E}_a, \mathcal{E}_b) + S(\mathcal{E}_b, \mathcal{E}_c) + S(\mathcal{E}_a, \mathcal{E}_c),$$

signifiant que le **trio** dans son intégralité apporte plus que n'importe quelle addition de paires isolées. Pour  $m > 3$ , on généralise ce principe : c'est la **coopération collective** qui prime.

En **théorie de l'information**, on retrouve cette distinction dans :

#### 62. Information mutuelle totale.

$$I(\mathbf{X}_1, \dots, \mathbf{X}_m) = \sum_{i=1}^m H(\mathbf{X}_i) - H(\mathbf{X}_1, \dots, \mathbf{X}_m),$$

laquelle ne fait pas le tri entre redondance et synergie réellement n-aire.

#### 63. Partial Information Decomposition (PID).

Ce formalisme sépare la part d'information partagée par tous (redondance) de la part spécifiquement **synergique**, c'est-à-dire l'information qui n'apparaît qu'en combinant l'ensemble complet des variables  $\mathbf{X}_1, \dots, \mathbf{X}_m$ . S'il émerge une synergie n-aire positive, on découvre un **surcroît** d'information inexistant dans tout sous-groupe plus restreint.

Pour le **DSL**, ces notions indiquent comment un cluster de taille  $m$  peut se constituer lorsqu'on détecte une synergie n-aire **positive** (ou dépassant un certain seuil).

## 2. Intégration de la synergie n-aire dans un réseau

Si les pondérations  $\omega_{i,j}$  portent sur des **liaisons binaires**, comment tirer parti d'une synergie n-aire ? Plusieurs pistes :

### 1. Approche factorisée.

Créer explicitement une pondération  $\omega_{k_1, \dots, k_m}$  reliant simultanément les entités  $\{\mathcal{E}_{k_1}, \dots, \mathcal{E}_{k_m}\}$ . Il faut alors manipuler un **hyper-graphe** (où une “arête” connecte plus de deux nœuds) ou stocker un **tenseur** de pondérations, ce qui devient coûteux si  $m$  croît.

### 2. Approche “bonus/malus”.

Reprendre le cadre binaire  $\omega_{i,j}$  en y ajoutant un **terme** lié à la synergie n-aire. Par exemple, pour trois entités  $\mathcal{E}_a, \mathcal{E}_b, \mathcal{E}_c$ , on écrit :

$$\omega_{a,b}(t+1) = \omega_{a,b}(t) + \eta [S(\mathcal{E}_a, \mathcal{E}_b) + \gamma S_3(\mathcal{E}_a, \mathcal{E}_b, \mathcal{E}_c) - \tau \omega_{a,b}(t)].$$

Ici,  $\gamma$  pèse la contribution de la synergie tripartite. Si  $\{\mathcal{E}_a, \mathcal{E}_b, \mathcal{E}_c\}$  forment un trio très coopératif, chaque lien binaire (a–b, b–c, a–c) reçoit un **surcroît** de renforcement, favorisant la cristallisation d'un micro-cluster à trois.

## 3. Émergence de micro- ou macro-clusters grâce à la synergie n-aire

Lorsqu'un groupe  $\{\mathcal{E}_{k_1}, \dots, \mathcal{E}_{k_m}\}$  manifeste une synergie n-aire élevée, on s'attend à :

1. **Renforcer** les liens binaires internes,
2. **Stabiliser** un sous-graphe “complet” ou presque, traduisant la cohésion du groupe,
3. Obtenir un **gain** global supérieur à la somme des seules paires.

Si, plus tard, un autre sous-groupe  $\{\mathcal{E}_{r_1}, \dots, \mathcal{E}_{r_q}\}$  se joint à ce collectif parce que la **synergie** entre ces deux ensembles élargis se révèle à son tour avantageuse, on peut assister à la **fusion** de clusters en un **macro-cluster** plus vaste. L'approche n-aire explique comment le DSL n'est pas limité au cas binaire : il peut trouver qu'un ensemble de 4, 5 ou 10 entités coopèrent de manière exceptionnelle, justifiant la formation d'un **module** émergent.

On peut y voir une **hiérarchie** (ou un treillis) :

- Les **feuilles** : synergies binaires  $\omega_{i,j}$ .
- Les **nœuds supérieurs** : synergies tri-partites, 4-partites, etc.
- Les **sommets** : macro-clusters englobant de grands ensembles.

L'**auto-organisation** d'un SCN peut alors être comprise comme un cheminement dans cet espace d'interactions multiples, valorisant les groupes où la **complémentarité** est nettement profitable.

## 4. Illustrations et implications

1. **Multimodalité.** Audio et visuel n'ont pas forcément une forte similarité, mais l'ajout d'une troisième modalité (ex. texte) peut révéler une synergie à trois entités. Ainsi, un macro-cluster se forme pour gérer ensemble la parole, l'image du locuteur et les informations projetées.
2. **Systèmes d'agents.** Trois robots (ou plus) peuvent constituer un **équipped** dont la coopération tri- ou multi-partite débloque une solution qu'aucune sous-combinaison plus simple ne parvenait à réaliser. Le DSL rend compte de cette nécessité de formation de “super-équipes”.
3. **Chimie ou biologie.** Deux composés chimiques A et B ne réagissent qu'imparfaitement, jusqu'à ce qu'on introduise un troisième C qui agit en **catalyseur**. On met alors en évidence une **synergie triple** (A–B–C) supérieure à toute combinaison restreinte.

## Conclusion

Si la **synergie binaire** constitue le socle même du SCN, la prise en compte de la **synergie n-aire** enrichit de manière considérable la capacité du DSL à **déceler** et **exploiter** des coopérations plus étendues. Les interactions à trois entités ou davantage peuvent révéler des **complémentarités** ou des **effets émergents** impossibles à discerner en se limitant aux paires. D'un point de vue pratique, le coût combinatoire de la gestion explicite de synergies n-aires peut se révéler élevé, mais des stratégies de type "bonus/malus" sur les liens binaires offrent des voies **approximatives** pour modéliser ces phénomènes.

Ainsi, la **synergie n-aire** confère au DSL un cadre globalement **plus puissant** et **expressif**, permettant de voir s'**auto-organiser** des ensembles coopératifs complexes et de mieux expliquer la naissance de **macro-clusters** fortement intégrés. Cette dimension s'ajoute à la dynamique temporelle (1.4.5) et aux interactions indirectes (1.4.6) pour compléter la **vision** d'un **réseau** éminemment **adaptatif**, ajustant ses liaisons au fil des découvertes sur la synergie de ses entités.

### 1.5. Pourquoi une Approche Synergique ?

Les sections précédentes ont mis en évidence la manière dont le **Deep Synergy Learning (DSL)**, à travers son **architecture générale** et son **auto-organisation** (voir sections 1.3 et 1.4), se distingue d'une approche classique. Pour mieux comprendre l'intérêt d'une telle démarche, il importe de répondre à une question cruciale : **pourquoi** choisir un paradigme "synergique" plutôt que de se contenter des réseaux de neurones profonds ou d'autres méthodes d'apprentissage ? Cette section 1.5 propose plusieurs raisons majeures :

4. **Avantages par Rapport aux Réseaux Neuraux Profonds** (1.5.1)
5. **Gestion Naturelle de la Multi-modalité** (1.5.2)
6. **Flexibilité vis-à-vis des Données Incomplètes ou Bruitées** (1.5.3)
7. **Potentiel d'Auto-Évolution et d'Adaptation Continue** (1.5.4)
8. **Réduction de la Dépendance à la Supervision Humaine** (1.5.5)
9. **Création de Représentations Riches et plus Interprétables** (1.5.6)
10. **Intégration de Dimensions Symboliques ou Cognitives** (1.5.7)

Chacun de ces points met en lumière les **bénéfices** d'un modèle qui valorise la **co-opération** dynamique entre entités, plutôt qu'une hiérarchie de couches figées. Nous allons tout d'abord (1.5.1) confronter la logique **synergique** du DSL à la structure rigide des **réseaux neuronaux profonds (Deep Learning)**, pour en dégager les avantages potentiels.

#### 1.5.1. Avantages par Rapport aux Réseaux Neuraux Profonds

Les **réseaux neuronaux profonds** (CNN, RNN, Transformers, etc.) ont assurément démontré leur **efficacité** dans de nombreux domaines. Cependant, le **DSL** propose une **philosophie** radicalement différente, susceptible d'apporter :

11. **Flexibilité Structurale**
12. **Plastique et Auto-Organisé**
13. **Réduction de la Spécialisation Rigide**
14. **Facilité d'Intégration Multi-entités**
15. **Apprentissage Continu**

Chacun de ces volets constitue un avantage important, que nous détaillons ci-après.

##### 1.5.1.1. Flexibilité Structurale

Dans un **réseau neuronal profond** classique, la **topologie** (nombre et type de couches, neurones, schémas de fusion de données) est entièrement **préétablie** avant l'apprentissage. Par exemple, un **CNN** (Convolutional Neural Network) organise les données selon des couches **convolutionnelles**, suivies éventuellement de **pooling** et de **couches fully-connected**, sans changement d'architecture au cours de l'entraînement. De même, les **Transformers** enchaînent des **blocs self-attention** répétés, toujours selon un canevas décidé en amont. Ces modèles **ne modifient** pas leur structure interne en cours de fonctionnement :

Nombre de couches, arrangement des blocs et synapses demeurent fixes pendant tout l'apprentissage.

Ils **n'autorisent** pas la création ou la **suppression** de neurones ou de liaisons et conservent un agencement identique, même si la **distribution** des données ou le **contexte** subit des modifications significatives.

À l'inverse, dans le **Deep Synergy Learning (DSL)**, on insiste sur la dynamique de ses **entités** et de leurs **liaisons**. Les pondérations  $\omega_{i,j}(t)$  entre deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  évoluent selon la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$ . Il est possible qu'un **lien** émerge (si la synergie grandit et dépasse un certain **seuil**), qu'il **disparaisse** (si l'intérêt coopératif s'avère trop faible), ou qu'il se **recrée** au cours du temps. La **flexibilité** topologique qui en résulte se formalise par des règles de mise à jour de la forme

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

où  $\eta$  désigne le **taux d'apprentissage** et  $\tau$  un **terme de régulation**. Quand la synergie  $S$  s'avère positive et assez élevée, la liaison se **renforce** ; dans le cas contraire, elle **s'affaiblit** et peut même s'annuler. Cette dynamique se révèle d'autant plus souple qu'on peut imposer une **parsimonie** via un seuil  $\omega_{\min}$  ou un **cap**  $\omega_{\max}$ , comme le rappelle la section 1.4.5.

La conséquence directe de cette **reconfiguration dynamique** tient dans la capacité du réseau à **s'adapter** en continu à l'arrivée de nouvelles données ou à un **changement** de distribution. Au lieu de maintenir invariablement le même chemin de traitement (comme dans un **CNN** figé ou un **Transformers** aux blocs intangibles), le **DSL** autorise une **reliance** directe entre certaines entités apparues pertinentes, tout en relâchant ou en supprimant d'autres connexions devenues inutiles. On peut ainsi écrire, de façon synthétique :

Flexibilité topologique  $\Rightarrow$  Capacité d'adaptation et résilience face à l'évolution du contexte.

Dès lors, si la distribution des données subit un changement progressif (par exemple, l'apparition d'une nouvelle modalité sensorielle), le **DSL** peut incorporer de nouvelles **entités**  $\mathcal{E}_{\text{nov}}$  et renforcer les liaisons  $\omega_{\text{nov},j}$  jugées porteuses d'une forte **synergie**. Simultanément, il peut affaiblir ou rompre des liens moins pertinents. C'est précisément cette **malléabilité** qui caractérise le **SCN** et le différencie de la rigidité structurelle des architectures profondes traditionnelles.

Ce **principe** de flexibilité structurelle jette les bases d'une **évolution** continue de la topologie, conforme à l'esprit d'**auto-organisation** présenté dans les sections 1.4.3 et 1.4.5. En ce sens, le **DSL** épouse davantage la **complexité** et la **variabilité** des environnements réels, tout en garantissant que seules les **connexions** profitables (au sens de la synergie) se **consolident** dans le temps.

### 1.5.1.2. Plasticité et Auto-Organisation

Dans les **réseaux de neurones** usuels, la phase d'apprentissage consiste principalement à **ajuster** les **poids** dans une **architecture** figée à l'avance. En revanche, le **Deep Synergy Learning (DSL)** adopte un parti pris plus **plastic** : il autorise la **topologie** du **Synergistic Connection Network (SCN)** à **évoluer** au cours du temps. Les liaisons  $\omega_{i,j}(t)$  qui peinent à démontrer une **synergie** positive ou dont la valeur reste trop faible peuvent **disparaître**, tandis que d'autres connexions, révélant un intérêt mutuel marqué, se **renforcent** jusqu'à forger de nouveaux chemins ou **clusters**.

Pour exprimer cette mise à jour, on recourt le plus souvent à une règle sous la forme

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

où  $\eta$  représente le **taux d'apprentissage**,  $\tau$  un terme de **régulation**, et  $S(\mathcal{E}_i, \mathcal{E}_j)$  la **synergie** (distance, similarité, information mutuelle...). Ainsi, chaque itération propose une **réévaluation** de la pertinence du lien  $\omega_{i,j}$ . Si la synergie demeure assez élevée, la liaison persiste ou se consolide ; sinon, elle tend vers zéro et peut être considérée comme **rompue** selon un seuil  $\omega_{\min}$  (cf. section 1.4.5 pour la gestion des seuils).

Cette **démarche** de régulation évoque la **plasticité synaptique** que l'on observe dans les **systèmes biologiques**. À l'image des neurones qui renforcent leurs connexions en cas de co-activation et en abandonnent d'autres, les entités du DSL remanient leur **graphe** de manière à privilégier les **coopérations** les plus fructueuses. Les **clusters** (sections 1.4.3 et 1.4.4) apparaissent alors **spontanément**, sans exiger de paramétrage explicite du nombre de groupes : ils émergent lorsque les pondérations internes d'un sous-ensemble d'entités dépassent certains niveaux de synergie, consolidant un **sous-réseau** fortement lié.

Cette aptitude à **auto-organiser** la structure s'avère avantageuse pour l'**adaptation** en continu. On peut en effet passer d'une tâche à l'autre ou intégrer une **nouvelle modalité** de données (image, audio, signal textuel) sans reconstruire l'architecture dans son intégralité, puisqu'il suffit d'ajouter des entités et de laisser les liaisons se former ou s'éteindre de façon autonome selon la **synergie** détectée. Les **clusters** demeurant utiles persistent, tandis que de **nouveaux** se créent si des **interactions** inédites se révèlent rentables. Le **réseau** conserve ainsi un **degré de flexibilité** qui lui permet d'**ajuster** sa **complexité** selon le **contexte**, proposant un fonctionnement plus souple qu'un modèle entièrement figé.

### 1.5.1.3. Réduction de la Spécialisation Rigide

Dans un **réseau profond** standard comme un **CNN** dédié à la reconnaissance visuelle, la structure se présente souvent de manière **verticalisée**, où les premières couches extraient des **bords**, puis des formes plus élaborées, jusqu'à la classification finale. Cette disposition rend la **réutilisation** des *features* complexes si l'on souhaite aborder d'autres tâches ou modalités, à moins d'effectuer un **fine-tuning** parfois lourd. Les couches initiales sont fixées dans une fonction très spécifique, et la **coopération** entre différents blocs (par exemple la fusion de vision et d'audio) se voit généralement cantonnée à un **niveau** imposé dans l'architecture ou à un **module** explicitement conçu pour la multimodalité.

Dans le cadre du **Deep Synergy Learning (DSL)**, la vision se veut plus **flexible**. Les **entités** composant le **Synergistic Connection Network (SCN)** incarnent chacune une **source d'information** ou un **bloc fonctionnel** susceptible d'être réutilisé dans d'autres contextes. La **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  détectée entre deux **entités**  $\mathcal{E}_i$  et  $\mathcal{E}_j$  ne se limite pas à un étage "plus haut" ou "plus bas" : toute entité peut, au fil des itérations, créer un **lien** avec toute autre, dès lors que la coopération leur procure un **gain** mesuré par la fonction  $S$ .

Un **exemple** concret peut être imaginé lorsqu'un **bloc**  $\mathcal{E}_{\text{bords}}$ , spécialisé dans la détection de **bords** visuels, observe soudain qu'il existe un **gain** d'associer ces **features** à un **bloc**  $\mathcal{E}_{\text{audio}}$  focalisé sur les **fréquences** sonores, notamment quand des corrélations entre mouvements de lèvres et signaux vocaux se manifestent. Au sein d'un **CNN** traditionnel, l'**intégration** de telles informations se produirait typiquement à un **niveau tardif** ou via un **module spécialisé**, ce qui limite la portée de la coopération. Dans le **DSL**, au contraire, la liaison  $\omega_{\text{bords, audio}}$  peut croître de manière **spontanée** si la **synergie** calculée (par exemple  $\exp(-\|\mathbf{x}_{\text{bords}} - \mathbf{x}_{\text{audio}}\|^2/\sigma^2)$ ) s'avère **élevée** et dépasse un certain **seuil**  $\omega_{\min}$  (section 1.4.5). Les deux blocs initient alors un **micro-cluster**, échangent leurs informations, et ainsi la **fusion** s'opère de manière autonome. Cette capacité à **tisser** des liens coopératifs entre entités diverge de la spécialisation rigide instaurée par des **pipelines** de couches fixes, et elle illustre la **souplesse** du DSL pour exploiter des **synergies** inattendues.

#### 1.5.1.4. Facilité d'Intégration Multi-Entités

Dans la plupart des **réseaux neuronaux profonds** classiques, il est usuel de concevoir un **design** architectural bien particulier pour **fusionner** plusieurs entrées issues de différentes modalités, comme une **image**, un **texte** et un **signal audio**. On trouve ainsi des approches où un **CNN** traite l'image, un **RNN** (ou un **Transformers**) interprète le texte, puis une **couche de fusion** se situe au sommet pour combiner les vecteurs latents. Cette disposition requiert, dans son principe, une intervention humaine afin de spécifier à quel niveau et selon quelles règles la **fusion** se produit.

Dans le **Deep Synergy Learning (DSL)**, la perspective est nettement plus **flexible**. Chaque **modalité** s'incarne dans une ou plusieurs **entités**  $\mathcal{E}_{\text{mod}}$  dédiées, que l'on peut définir comme un bloc perceptif ou un ensemble de **features** cohérent. Les **liens**  $\omega_{i,j}$  se **construisent** de manière autonome, selon la **synergie** mesurée entre les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . On peut exprimer cette dynamique à l'aide de la règle d'adaptation :

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

où  $\eta$  désigne le **taux d'apprentissage**,  $\tau$  un **terme de régulation**, et  $S(\mathcal{E}_i, \mathcal{E}_j)$  la **synergie** (section 1.4.4). Cette règle fait en sorte que toute **coopération** mutuellement avantageuse s'**intensifie** si elle contribue à un gain, alors que les associations moins pertinentes se **dissipent**. Dans un **contexte** multimodal, l'image, le texte et l'audio sont autant d'entités  $\{\mathcal{E}_{\text{image}}, \mathcal{E}_{\text{texte}}, \mathcal{E}_{\text{audio}}\}$  susceptibles de découvrir des **corrélations** bilatérales ou trilatérales, sans qu'une **couche de fusion** imposée par un concepteur soit nécessaire.

Lorsque la **synergie** entre ces modalités se révèle conséquente, les pondérations  $\omega_{\text{image, texte}}$ ,  $\omega_{\text{image, audio}}$  ou  $\omega_{\text{texte, audio}}$  croissent, formant de manière **spontanée** un **cluster** multimodal où les entités coopèrent directement. Cette **auto-organisation** se renforce si l'association apporte un bénéfice mesurable (par exemple, une performance améliorée sur une tâche de classification ou une plus grande robustesse). Inversement, si une modalité  $\mathcal{E}_{\text{mod}}$  n'apporte pas d'information utile ou se révèle trop bruitée, la **synergie** reste faible et la liaison ne s'établit pas, ou se rompt rapidement.

Une autre conséquence réside dans la **capacité** du réseau à **adapter** en continu les entités multimodales, autorisant la **formation** et la **dissolution** d'un **cluster** au gré du contexte. Cette propriété est en net contraste avec l'idée traditionnelle d'une couche de fusion figée où toutes les modalités se rejoignent selon un schéma unique. Le **DSL** propose donc une **répartition** dynamique de l'intégration multimodale, dans laquelle les **entités** se reconfigurent en fonction de l'**opportunité** d'apprentissage.

Au final, la **facilité d'intégration multi-entités** découle directement de la manière dont les **liaisons synergiques** sont définies et mises à jour. Chaque modalité, qu'elle concerne le **visuel**, le **textuel**, le **sonore** ou un **autre flux**, est **libre de découvrir** et de **former** des **connexions** directes avec n'importe quel **bloc** complémentaire, dès lors qu'une **valeur ajoutée** en résulte. Le **réseau** opère ainsi sous une forme d'**auto-organisation** (sections 1.4.3 et 1.4.5), permettant d'exploiter la diversité des **modalités** sans nécessiter le design d'un unique **module** de fusion, et garantissant une **adaptation** continue à la **variabilité** de l'environnement.

#### 1.5.1.5. Apprentissage Continu

Dans de nombreux **réseaux neuronaux profonds**, l'entraînement se déroule dans un cadre essentiellement statique. On se contente généralement d'effectuer une rétropropagation sur un **ensemble de données** figé, et il s'avère difficile de réaliser un **apprentissage continu** sans retomber dans le phénomène de **catastrophic forgetting** au moment d'acquérir de nouvelles tâches ou de faire face à un changement de distribution. De plus, la structure même du réseau (couches, neurones) demeure immuable pendant tout l'apprentissage, ce qui limite considérablement la possibilité d'**évoluer** au fil du temps.

À l'inverse, le **Deep Synergy Learning (DSL)** s'appuie sur un **mécanisme** de **pondérations adaptatives** (discuté en section 1.4.5) dans lequel chaque liaison  $\omega_{i,j}(t)$  peut se créer, se renforcer ou disparaître selon la **synergie** qu'elle procure. Cette propriété ouvre la voie à un **apprentissage continu** plus fluide, car le **réseau** peut accueillir de nouvelles **entités**  $\mathcal{E}_{\text{nouv}}$  chaque fois qu'un nouveau flux de données ou une nouvelle modalité fait son apparition. Les **liaisons**  $\omega_{\text{nouv}, j}$  se forment si elles sont jugées bénéfiques, tandis que les anciens **clusters** jugés toujours utiles demeurent stables. Ainsi, l'**oubli** des acquis se trouve **atténué**, puisque les **connections** consolidées par le passé ne sont pas supprimées tant qu'elles conservent une synergie positive. Les entités correspondant aux tâches ou domaines précédents (sections 1.4.3 et 1.4.4) ne disparaissent pas, et les sous-groupes de coopération qui s'étaient formés peuvent persister ou se réactiver ultérieurement.

Une telle **flexibilité** convient particulièrement aux **environnements dynamiques**, où le **réseau** doit sans cesse **apprendre** de nouvelles classes d'objets, de nouvelles langues, ou s'adapter à des évolutions de contexte sans recommencer l'**entraînement** depuis zéro. Le **DSL** se distingue alors de la rigidité caractéristique d'un **réseau profond** standard, qui impose généralement de réinitialiser ou de "fine-tuner" un ensemble de couches fixes lors de l'introduction d'un nouveau domaine. La possibilité de **reconfigurer** la topologie et d'**absorber** ou de **cesser** d'utiliser certaines sources de données fait du **SCN** un cadre mieux adapté à un **apprentissage Lifelong** ou **continu**, tout en préservant la robustesse et la puissance d'un paradigme sub-symbolique.

## Conclusion

En comparaison directe avec les **réseaux neuronaux profonds**, l'**approche synergique** du DSL propose :

- **Une topologie évolutive**, au lieu d'une hiérarchie figée,
- **Une plasticité** inspirée (partiellement) de la biologie, plutôt qu'un pipeline statique,
- **Une modularité** et **une fusion** multimodale plus naturelles, sans imposer de couches de fusion,
- **Un apprentissage continu** mieux géré, permettant l'ajout ou la suppression d'entités sans tout réapprendre.

Il ne s'agit pas de nier la **puissance** des réseaux profonds : ils restent extrêmement performants lorsqu'on dispose de gros volumes de données annotées et qu'on accepte une architecture prédéfinie. Mais le DSL entend dépasser les limites liées à la **spécialisation**, au **statisme** et à la **dépendance** de l'apprentissage neuronal classique, ouvrant la voie à une **IA adaptative** et plus **ouverte** à la co-opération des informations.

Dans les sous-sections suivantes (1.5.2 à 1.5.7), nous approfondirons d'autres avantages majeurs, comme la **gestion naturelle de la multi-modalité** (1.5.2), la **flexibilité** face aux données partielles (1.5.3), l'**auto-évolution** (1.5.4), la **réduction** de la supervision (1.5.5), la **création** de représentations plus **riches** (1.5.6) et l'**intégration** de dimensions **symboliques** (1.5.7). L'ensemble consolidera l'idée qu'une **approche synergique** offre un panel d'atouts pour concevoir des systèmes apprenants plus **évolutifs**, **modulaires** et **généraux** que ne le permettent les architectures entièrement figées du deep learning traditionnel.

### 1.5.2. Gestion Naturelle de la Multi-modalité

Une des motivations fortes qui a conduit à l'élaboration du **Deep Synergy Learning (DSL)** est la possibilité d'**intégrer** plusieurs types de données (*images, sons, textes, capteurs variés, etc.*) de manière à ce qu'elles **coopèrent** au lieu de simplement être fusionnées dans une couche dédiée. Dans les réseaux neuronaux profonds classiques, la multi-modalité impose souvent de **concevoir** des modules spécifiques (p. ex. un CNN pour l'image, un RNN ou un Transformer pour le texte, un autre réseau pour l'audio), puis de **fusionner** ces modules dans une partie supérieure de l'architecture. Cette approche, quoique efficace dans de nombreux cas, reste relativement **rigide** : la fusion se fait à un niveau imposé et ne varie guère en fonction du contexte ou de la dynamique interne.

Le **DSL**, au contraire, encourage une **intégration plus organique** : chaque modalité est représentée par une (ou plusieurs) **entité(s)**  $\mathcal{E}_{\text{mod}}$ , qui cherchent spontanément à établir ou à rompre des liens avec des entités d'autres modalités, suivant la **synergie** constatée. Cette sous-section (1.5.2) détaille :

16. **Pourquoi** la multi-modalité est "naturellement" absorbée par le DSL,
17. Comment les entités **visuelles, auditives, textuelles** (par ex.) peuvent **former** des clusters multimodaux,
18. Quels **avantages** cette gestion apporte (robustesse, découvertes de liens inattendus, etc.).

#### 1.5.2.1. Les Entités comme Vecteurs ou Blocs Multimodaux

Dans le **Deep Synergy Learning (DSL)**, chaque **entité**  $\mathcal{E}_i$  peut se spécialiser dans un type de données particulier ou combiner déjà plusieurs **descripteurs**. Il n'y a pas l'obligation, comme dans un réseau multimodal classique, d'assembler deux flux (image, audio, etc.) au niveau d'une couche imposée ou dans des branches séparées. Au lieu de cela, toutes les entités coexistent dans un **même "espace"** ou **graphe**  $\{\mathcal{E}_1, \mathcal{E}_2, \dots, \mathcal{E}_n\}$ . On y retrouve des entités  $\mathcal{E}_{\text{image}}, \mathcal{E}_{\text{audio}}, \mathcal{E}_{\text{texte}}$ , ou des blocs déjà **multimodaux** combinant différents **features**. Il n'existe pas de **pyramide** de couches fixes, comme un **CNN** qui détecterait d'abord les bords, puis les textures, puis les objets. Au contraire, le **DSL** insiste sur le principe d'**auto-organisation**, faisant que toute entité  $\mathcal{E}_i$  peut se **lier** à toute autre  $\mathcal{E}_j$  dès lors qu'une **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  significative est détectée. La mise à jour de la pondération  $\omega_{i,j}(t)$  suit la règle discutée en section 1.4.5, et si la coopération apporte un gain, on obtient un renforcement de la liaison  $\omega_{i,j}$ . Ainsi, une entité de type **texte**  $\mathcal{E}_{\text{txt}}$  n'est pas cantonnée à se fusionner uniquement avec une entité **image**  $\mathcal{E}_{\text{img}}$  dans un étage supérieur prédéfini : la **fusion** peut advenir à **n'importe quel moment**,

entre n'importe quelles entités, pourvu qu'une **valeur ajoutée** soit constatée. Cette démarche favorise la **formation** naturelle de **clusters** multimodaux là où c'est nécessaire, et permet aux entités déjà mixtes (par exemple combinant audio et vision) de se relier également à d'autres (texte, capteurs...), rendant le réseau plus **souple** et mieux à même de gérer la **pluralité** des sources de données.

### 1.5.2.2. Formation de Clusters Multimodaux

Le **Deep Synergy Learning (DSL)** emploie une règle de mise à jour décrite dans la section 1.4.5, qui se formalise ainsi :

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

Lorsque, par exemple, une **entité visuelle**  $\mathcal{E}_{\text{visuelle}}^a$  (extraction d'images) et une **entité auditive**  $\mathcal{E}_{\text{auditive}}^b$  (analyse du spectre sonore) constatent une **synergie** satisfaisante, la liaison  $\omega_{a,b}$  se **renforce** progressivement. Les sections 1.4.3 et 1.4.4 évoquent déjà ce principe de consolidation, où toute **coopération** profitable se traduit par une augmentation de la pondération. Au fil des itérations, d'autres entités, qu'elles soient textuelles ou associées à une autre modalité, peuvent se **greffer** à ce duo si elles y perçoivent elles aussi un **gain**. S'organise alors un **cluster** multimodal  $\{\mathcal{E}_{\text{vis}}, \mathcal{E}_{\text{aud}}, \mathcal{E}_{\text{txt}}\}$ , destiné, par exemple, à la reconnaissance d'événements conjoints dans une vidéo, en y associant le son et des sous-titres.

Un exemple important survient lors de l'analyse synchronisée **vidéo + audio**. Une entité  $\mathcal{E}_{\text{CNN}}$ , spécialisée dans la détection d'objets ou de mouvements visuels, et une entité  $\mathcal{E}_{\text{Audio}}$ , orientée vers les fréquences sonores, découvrent une **corrélation** entre le contenu visuel (mouvements de l'orateur) et des sons spécifiques. Si la **fonction de synergie**  $S(\mathcal{E}_{\text{CNN}}, \mathcal{E}_{\text{Audio}})$  affiche une valeur élevée, la pondération  $\omega_{\text{CNN, Audio}}$  croît. Il en résulte la formation d'un **sous-réseau** constitué de ces entités, apte à identifier les vidéos où un son précis se déclenche simultanément avec un geste ou un mouvement labial.

Un autre scénario se rencontre dans la **fusion texte + image**. Si une entité  $\mathcal{E}_{\text{visuel}}$  extrait les objets visibles dans une scène et qu'une entité  $\mathcal{E}_{\text{langage}}$  génère des phrases descriptives (via un modèle de type RNN ou Transformers), la **co-information** qu'elles partagent peut montrer qu'en les associant, on obtient une légende automatique bien plus fidèle. Cette plus-value se reflète dans la liaison  $\omega_{\text{visuel, langage}}$ , qui se **solidifie** de manière autonome. Le **cluster**  $\{\mathcal{E}_{\text{visuel}}, \mathcal{E}_{\text{langage}}\}$  se spécialise alors dans la génération de légendes, sans imposer de couche de fusion prédéterminée.

On peut enfin imaginer un **cluster hybride** regroupant à la fois la "capture de mouvement" via des capteurs inertiels, la "reconnaissance d'images" au moyen d'une entité visuelle, et l'"analyse du spectre sonore". Ce **triplet** se cristallise dès lors qu'il existe une forte complémentarité pour identifier des événements qui combinent aspects visuels, audio, et déplacements corporels. Le **SCN** encourage ainsi la création de tels **clusters** dès que les synergies apparaissent bénéfiques, ce qui permet au réseau de s'adapter aux environnements multimodaux et de tirer le meilleur parti de chacune des sources disponibles.

### 1.5.2.3. Avantages de la Co-Organisation Multimodale

Dans un **réseau neuronal** habituel, la fusion des différents **flux** (par exemple, l'image et le son) est généralement décidée à l'avance, souvent dans les étages supérieurs ou dans une **couche intermédiaire** prédéfinie. Le **Deep Synergy Learning (DSL)** introduit une perspective différente : chaque **entité**  $\mathcal{E}_i$  peut spontanément **rechercher** des partenaires synergiques, **évaluer** si la coopération améliore les performances ou apporte un gain en information mutuelle, et **créer** des liaisons directes  $\omega_{i,j}$  sans qu'une couche de fusion imposée soit nécessaire. Cette démarche favorise l'apparition de **combinaisons** inhabituelles mais profitables, comme l'association de l'analyse audio, de la mesure de température et de l'intensité lumineuse, si un cluster pertinent émerge.

Concrètement, la **règle** de mise à jour

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

permet à chaque entité de constater, de façon autonome, si sa **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  s'avère suffisamment élevée pour justifier une **collaboration**. Ainsi, il n'existe pas d'étage unique où l'on agrégerait tous les flux, mais plutôt un **réseau** dont la topologie évolue pour intégrer toute coopération bénéfique.

Lorsqu'une modalité est **perturbée** ou moins fiable (par exemple l'audio, soumis à un **bruit** important ou une panne de capteur), la synergie associée  $\omega_{\text{aud.}}$  s'affaiblit, car les entités n'y trouvent plus de **valeur ajoutée**. Le **cluster** multimodal se réorganise donc en renforçant d'autres connexions, comme celles reliant la vision, le texte ou des capteurs alternatifs. Le système bénéficie ainsi d'une **résilience** accrue : il ne dépend pas d'un pipeline figé ni d'un module unique pour chaque modalité, mais s'autorégule en continu pour préserver une **robustesse** globale face aux changements du contexte ou aux défaillances de certaines sources.



#### 1.5.2.4. Éléments Mathématiques : Synergie “Multi-Modal”

Dans le **Deep Synergy Learning (DSL)**, la mesure de **synergie** entre différentes **modalités** (comme l'audio, la vision ou le texte) peut prendre plusieurs formes selon le type de données ou la définition souhaitée de la **co-opération**. On considère deux grandes approches : l'usage d'une **co-information** (ou multi-information) tirée de la théorie de l'information, et le recours à une **distance** ou **similarité** inter-modale dans un espace latent commun.

##### A. Mesure de co-information

Lorsque l'on manipule un ensemble de modalités  $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_m$ , il est possible d'estimer une **co-information** ou **mutual information** globale :

$$S(\mathbf{X}_1, \dots, \mathbf{X}_m) = \text{MI}(\mathbf{X}_1, \dots, \mathbf{X}_m).$$

Dans certains cas, la **Partial Information Decomposition (PID)** permet de séparer la part strictement *synergique* de la part *redondante* ou *unique*. Dans le **DSL**, on s'intéresse à l'**analyse n-aire** au sens où si la **conjonction** de deux (ou plus) modalités, par exemple  $\mathbf{X}_{\text{visuelle}}$  et  $\mathbf{X}_{\text{auditive}}$ , produit une **information** qu'aucune des modalités ne détenait isolément, alors on en conclut un **gain** pour la **synergie**. Ce **gain** s'exprime par un **renforcement** des pondérations : si l'usage conjoint d'un flux visuel et d'un flux audio s'avère utile, la liaison  $\omega_{\text{vis,aud}}$  augmente selon la règle de mise à jour (section 1.4.5). L'**auto-organisation** du réseau valorise ainsi la coopération entre modalités lorsqu'elle apporte un surcroît d'information ou de performance.

##### B. Distance ou Similarité inter-modale

En alternative à la co-information, on peut quantifier la **synergie** via une **distance** (ou **similarité**) entre les modalités, à condition de se situer dans un **espace latent** commun. Dans la pratique, cela implique généralement les étapes suivantes :

(1) Embeddings partagés.

On forme une représentation vectorielle  $\mathbf{z}_{\text{vis}}$  pour la partie **visuelle** et  $\mathbf{z}_{\text{aud}}$  pour la partie **audio**, toutes deux dans  $\mathbb{R}^d$ . Ces embeddings peuvent provenir d'un réseau d'apprentissage dédié ou être appris conjointement.

(2) Distance ou similarité.

On calcule une **norme**  $\|\mathbf{z}_{\text{vis}} - \mathbf{z}_{\text{aud}}\|$  ou une **similarité cosinus**  $\langle \mathbf{z}_{\text{vis}}, \mathbf{z}_{\text{aud}} \rangle$ . Cette quantité rend compte de la **proximité** ou de la **complémentarité** entre les deux modalités une fois projetées dans l'espace latent.

(3) Définition de la synergie.

En inversant ou en normalisant la distance, on obtient un **score** de **coopération** :

$$S(\mathbf{z}_{\text{vis}}, \mathbf{z}_{\text{aud}}) = \exp(-\|\mathbf{z}_{\text{vis}} - \mathbf{z}_{\text{aud}}\|^2 / 2\sigma^2) \quad \text{ou} \quad 1 / (1 + \|\mathbf{z}_{\text{vis}} - \mathbf{z}_{\text{aud}}\|^2),$$

ou encore en recourant à une **similarité** normalisée dans  $[0,1]$ . Cette mesure, insérée dans la formule de mise à jour des pondérations  $\omega_{i,j}(t+1) = \omega_{i,j}(t) + \dots$ , renseigne sur la **pertinence** qu'il y a à **connecter** et **coordonner** ces deux modalités.

##### Synthèse et implication dans le DSL

Le **DSL** n'impose pas de définition unique de la synergie ; toute **fonction** mesurant la **coopération** peut s'appliquer. La **co-information** met en évidence des **complémentarités** complexes (non linéaires), en évaluant l'information que seules deux (ou plusieurs) modalités, **combinées**, parviennent à extraire. De son côté, la **distance** (ou la **similarité**) dans un **espace latent** offre un calcul plus direct et souvent moins coûteux, permettant d'établir rapidement une **coopération** si les embeddings s'avèrent proches ou orientés de façon compatible.

Dans tous les cas, on reste dans l'esprit du **DSL** : la synergie inter-modale agit comme un **signal** indiquant que les entités multimodales gagneraient à se **lier** et à former un cluster dédié. Cette logique s'applique à deux entités (binaire), mais peut s'étendre à des groupes plus vastes (section 1.4.7) lorsque plusieurs **modalités** s'enrichissent mutuellement. Dans une perspective **multimodale**, on voit ainsi se constituer des **clusters** audio-vidéo-texte, ou audio-vision-capteurs, sans qu'un concepteur doive imposer explicitement à quel niveau se fait la **fusion**. Le **SCN** se reconfigure et s'**auto-organise**, renforçant les **liens** rentables au sens de la synergie globale.

#### 1.5.2.5. Perspectives : Plus de Fluidité, Plus de Découvertes

Dans un **Deep Synergy Learning (DSL)**, la fusion multimodale n'est pas confinée à une couche ou un module imposé à l'avance. Elle repose sur l'aptitude de chaque **entité** à forger des liens **synergiques** dès lors que la coopération procure un **gain**. Cette configuration autorise plusieurs évolutions particulièrement souples.

Un premier aspect concerne la **création spontanée** de **clusters multimodaux**. Lorsqu'un événement sollicite simultanément divers flux, par exemple un son particulier, un motif visuel distinctif et un mot-clé textuel, le **DSL** favorise la formation d'un **cluster** auto-organisé combinant toutes ces sources. Il n'existe pas l'exigence de décider, dans la conception du réseau, à quel étage ou sous-ensemble la

fusion doit survenir : le **Synergistic Connection Network** se charge de **détecter** les entités pertinentes, puis de renforcer leurs liaisons  $\omega_{i,j}$  si la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  s'avère élevée. Cette auto-organisation, évoquée en section 1.4.3, rend la fusion bien plus fluide et adaptative.

Un deuxième enjeu apparaît lorsque des **modalités** ou sources de données nouvelles se présentent. Si un capteur inédit vient produire un nouveau flux, il suffit d'introduire une entité  $\mathcal{E}_{\text{capteur}}$  correspondante. Les pondérations  $\omega_{\text{capteur},x}$  se mettent alors à jour selon la même loi que précédemment (section 1.4.5). Si les interactions avec des entités déjà existantes procurent un réel **bénéfice** à l'apprentissage ou à la représentation, la **synergie** grimpe ; sinon, le lien demeure faible et finit par s'éteindre. L'essentiel tient dans le fait qu'aucune **reconstruction** de l'architecture globale n'est nécessaire : le réseau s'**auto-adapte**, créant et dissolvant les connexions au gré des "opportunités" que confère le nouveau flux.

Un troisième point souligne la **réutilisation** des entités spécialisées. Dans un paradigme classique, un module dédié, par exemple à l'analyse de fréquences sonores, risque de ne pouvoir servir qu'à une tâche unique (identification de parole ou détection de musique). Au contraire, dans le **DSL**, si une entité  $\mathcal{E}_{\text{audio}}$  sait détecter des éléments discriminants dans le spectre sonore, elle peut être **sollicitée** par des **clusters** différents : on peut ainsi la retrouver dans un sous-réseau chargé de la reconnaissance de mots clés, un autre consacré à la détection de bruits inhabituels, ou encore un qui s'occupe d'associer des fragments de parole aux mouvements de lèvres. Chaque fois que la **synergie** le justifie, la liaison  $\omega_{\text{audio},...}$  se renforce dans le cluster concerné, sans imposer de duplication de la fonctionnalité ni de modification d'une couche fixe.

L'ensemble de ces mécanismes assure une **fluidité** et un **potentiel d'exploration** supérieurs par rapport aux modèles multimodaux figés. Le **DSL** favorise l'**émergence** et la **dissolution** dynamiques de clusters multimodaux, l'**intégration** graduelle de modalités nouvelles, et la **mutualisation** de blocs spécialisés au service de plusieurs tâches. Cette architecture, loin d'être imposée, se déploie au fil des itérations par simple adaptation des pondérations  $\{\omega_{i,j}(t)\}$ , conduisant à une plus grande **richesse** de découvertes et de combinaisons entre entités.

## Conclusion

La **multi-modalité** constitue un **terrain privilégié** pour apprécier la **valeur ajoutée** du Deep Synergy Learning. Là où un **réseau neuronal** classique cloisonne les modalités dans des branches séparées, puis les fusionne **sur** commande, le DSL propose :

- Une **co-intégration libre** des entités,
- Une **dynamique** où le réseau détecte **de lui-même** quelles modalités se complètent,
- Une **robustesse** accrue lorsque certaines modalités sont indisponibles ou bruitées,
- Une **exploration plus approfondie** des interactions possibles (même celles qu'on n'avait pas anticipées).

C'est pourquoi, dans l'optique d'une IA **multi-sensorielle** ou traitant des **sources hétérogènes**, le DSL offre un **cadre naturel** et **organique** pour favoriser les **synergies**. On verra dans les points suivants (1.5.3 à 1.5.7) d'autres aspects tout aussi clés (comme la flexibilité face aux données incomplètes, la réduction du besoin de supervision, ou la possibilité d'inclure des dimensions symboliques).

### 1.5.3. Flexibilité vis-à-vis des Données Incomplètes ou Bruitées

Outre la capacité à gérer de multiples flux (voir 1.5.2), l'un des atouts majeurs du **Deep Synergy Learning (DSL)** réside dans sa **tolérance** accrue face aux **données incomplètes** ou **fortement bruitées**. Dans les approches d'apprentissage traditionnelles (réseaux neuronaux profonds inclus), on cherche souvent à **normaliser** ou **compléter** les données manquantes, voire on opte pour un **prétraitement** lourd afin de filtrer les bruits. Le DSL, de par sa **structure auto-organisée**, autorise un réseau à composer **localement** avec les lacunes et à **réajuster** ses connexions de manière à s'appuyer sur les sources les plus fiables.

Cette section (1.5.3) met en avant les principes qui confèrent au DSL sa **robustesse** et sa **souplesse** quand on fait face à des entrées partielles, hétérogènes ou bruitées.

#### 1.5.3.1. Rôle de l'Auto-Organisation dans la Gestion du Bruit

Dans le **Deep Synergy Learning (DSL)**, les entités  $\{\mathcal{E}_1, \dots, \mathcal{E}_n\}$  s'**agglomèrent** en **clusters** (tels que décrits en section 1.4.3) lorsque leurs synergies bilatérales  $S(\mathcal{E}_i, \mathcal{E}_j)$  atteignent un certain **niveau**. L'auto-organisation agit alors comme un **mécanisme** filtrant : si une entité  $\mathcal{E}_k$  se révèle trop **bruyante**, ou plus généralement peu profitable (synergie insuffisante, voire négative, avec les autres), ses connexions  $\omega_{k,}$  **dépérissent** naturellement, selon la **mise à jour** :

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

Si la synergie  $S(\mathcal{E}_k, \mathcal{E}_j)$  reste **faible** ou négative, la liaison  $\omega_{k,j}$  diminue itération après itération jusqu'à se rapprocher d'un **seuil**  $\omega_{\min}$  en deçà duquel elle est jugée **inexistante**. L'entité  $\mathcal{E}_k$  tend dès lors à **s'isoler** de la structure principale et ne contribue plus de manière significative aux décisions ou au regroupement de clusters, limitant l'**impact** du bruit ou des mesures peu fiables sur l'ensemble du **Synergistic Connection Network**. Cette propriété confère au **DSL** une **robustesse** notable, car il n'est pas nécessaire de filtrer d'emblée les données douteuses : c'est la **dynamique** des pondérations qui effectue ce filtrage de manière locale et autonome.

De façon analogue, une entité ne possédant que des **données incomplètes** (par exemple, un capteur qui ne fonctionne que par intermittence) peut tout de même établir des **liens forts** avec d'autres entités si la partie (même partielle) de ses observations est **pertinente** et apporte un gain synergique réel. L'apprentissage ne condamne pas a priori les capteurs partiellement défaillants : il évalue la **valeur ajoutée** qu'ils procurent (section 1.4.4 sur la définition de  $S$ ), et **renforce** ou **dissout** les liaisons en conséquence. Ainsi, le **DSL** teste localement l'**efficacité** des données partielles et décide, via la seule dynamique adaptative de  $\omega_{k,j}(t)$ , si l'entité doit être conservée dans un cluster ou rester en marge du réseau.

Ce procédé évite la nécessité d'une **stratégie** spécifique pour rejeter le **bruit** ou manipuler les **données manquantes**. Le **système** s'auto-régularise, en quelque sorte, grâce à la diminution des pondérations non rentables, ce qui se traduit par une **atténuation** de l'influence des sources imprécises. L'**auto-organisation** réalise donc une forme de **gestion du bruit** distribuée, où chaque entité (ou module) se maintient ou s'éclipse selon la **synergie** réellement engendrée avec les autres composantes du **SCN**, évitant ainsi de nuire à la **performance** globale.

### 1.5.3.2. Interprétation dans un Cadre Bruité

Lorsqu'une entité  $\mathcal{E}_k$  se trouve exposée à un **bruit** important dans ses représentations  $\mathbf{x}_k$ , sa **similarité** ou son **information mutuelle** avec les autres entités demeure faible ou fluctuante. On peut considérer par exemple la fonction

$$S(\mathcal{E}_k, \mathcal{E}_j) = \frac{1}{1 + \|\mathbf{x}_k - \mathbf{x}_j\|^2}.$$

Si  $\mathbf{x}_k$  varie de manière aléatoire (bruit élevé), alors  $\|\mathbf{x}_k - \mathbf{x}_j\|$  reste souvent grande ou instable, réduisant ainsi la valeur de  $S(\mathcal{E}_k, \mathcal{E}_j)$ . En se référant à la loi de mise à jour des pondérations (section 1.4.5), on voit que

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

La valeur de  $\omega_{k,j}(t)$  décroît donc si la **synergie**  $S(\mathcal{E}_k, \mathcal{E}_j)$  ne compense pas suffisamment le terme de régulation  $\tau \omega_{k,j}(t)$ . Cette **diminution** progressive des liaisons  $\omega_{k,j}$  conduit l'entité  $\mathcal{E}_k$  à **s'isoler** si le bruit la rend incohérente vis-à-vis des entités du cluster. Il n'est pas nécessaire d'introduire un algorithme d'exclusion particulier : le **DSL** gère ce phénomène de façon **auto-organisée**, en laissant le lien s'affaiblir jusqu'à potentiellement s'éteindre lorsque

$$\omega_{k,j}(t) < \omega_{\min}.$$

Dans l'hypothèse où, plus tard,  $\mathcal{E}_k$  retrouve une plus grande **fiabilité** (par exemple si un paramètre interne  $\mathbf{s}_k(t)$  se recalibre, ou si le flux de données s'assainit), on observe que la **distance**  $\|\mathbf{x}_k - \mathbf{x}_j\|$  peut redevenir modérée, augmentant  $S(\mathcal{E}_k, \mathcal{E}_j)$ . La mise à jour  $\omega_{k,j}(t+1)$  peut alors reprendre une tendance à la hausse, recréant ou solidifiant des liaisons  $\omega_{k,j}$ . Cette **réversibilité** illustre l'une des forces du **Synergistic Connection Network** : une entité qui s'était tenue à l'écart en raison d'un bruit excessif peut retrouver **sa place** dès que ses **observations** redeviennent pertinentes, sans nécessiter la restructuration globale du réseau. L'entité revient alors dans le **cluster**, démontrant la **capacité** du DSL à **s'adapter** en continu à la qualité fluctuante des données.

### 1.5.3.3. Données Partielles : Complétion Progressive via Synergie

Dans de nombreux **scénarios** pratiques, certaines entités ne disposent que d'une **fraction** de leurs attributs. Par exemple, dans un cadre médical, un **patient** peut présenter des **données** incomplètes : quelques tests sanguins manquants, une imagerie non réalisée, ou un historique lacunaire. Les approches neuronales classiques exigent souvent une **imputation** ou un **remplissage** préalable des valeurs absentes, ce qui nécessite des hypothèses ou des méthodes de substitution. Il n'est pas rare non plus que l'on décide simplement d'**écarter** les exemples incomplets pour ne pas perturber l'apprentissage.

Dans le **Deep Synergy Learning (DSL)**, une telle entité — par exemple  $\mathcal{E}_{\text{patient}}$  — est autorisée à **coopérer** partiellement avec d'autres entités. Si l'entité patient ne possède que certaines **mesures** (tests sanguins, variables cliniques) mais en ignore d'autres, elle peut néanmoins évaluer sa **synergie** avec, par exemple, d'autres patients  $\mathcal{E}_{\text{patient}}$  ou des **variables globales** (moyennes de cohortes, règles médicales) en se basant sur les **dimensions** ou **attributs** qu'elle détient réellement. Une fonction de synergie  $S(\mathcal{E}_k, \mathcal{E}_j)$  adaptée aux **dimensions communes** ou à une distance partielle peut alors guider la mise à jour de la pondération :

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

Si, malgré des données incomplètes, l'entité  $\mathcal{E}_k$  dégage une **valeur ajoutée** non négligeable pour le réseau (par exemple, la portion de tests sanguins disponibles est très informative), ses **liaisons**  $\omega_{k,j}$  avec d'autres entités  $\mathcal{E}_j$  se **consolident** progressivement. Cela peut se traduire par la **constitution** d'un **cluster** rassemblant d'autres patients, ou bien des entités spécialisées (un bloc fonctionnel détectant une tendance clinique). L'**incomplétude** des données n'empêche pas la **coopération** : seul importe le **gain** mesuré sur les attributs effectivement partagés.

Au fur et à mesure que l'entité  $\mathcal{E}_k$  acquiert (ou reconstitue) de nouveaux attributs manquants, il est envisageable que sa **distance** ou sa **similarité** avec d'autres entités évolue, rehaussant la pondération  $\omega_{k,j}$ . Le réseau y trouve un bénéfice : la réévaluation de la synergie incorpore cette **nouvelle information**, favorisant le **réajustement** des liens. On évite ainsi l'**exclusion** systématique des cas incomplets ou la nécessité d'une **imputation** globale : le **DSL** se contente de « tester » la **collaboration** sur la partie disponible des données. Cette logique de **liaison partielle** confère au **DSL** une **souplesse** précieuse pour manipuler des **jeux de données** irréguliers, tout en conservant la capacité d'un **regroupement** (cluster) auto-organisé autour des **similarités** ou **complémentarités** réellement détectées.

#### 1.5.3.4. Exemples Concrets et Bénéfices

Dans un **environnement industriel**, certains capteurs (de température, de pression...) se révèlent souvent **défaillants** ou trop **bruités**. Dans un **Deep Synergy Learning (DSL)**, les entités correspondantes  $\mathcal{E}_{\text{temp}}$  ou  $\mathcal{E}_{\text{press}}$  voient leurs **pondérations**  $\omega_{\text{temp},j}$ ,  $\omega_{\text{press},j}$  diminuer dès lors que la **synergie** avec d'autres capteurs (comme  $\mathcal{E}_{\text{vibration}}$  ou  $\mathcal{E}_{\text{débit}}$ ) ne justifie plus la coopération. Le **cluster** principal, dédié par exemple à la **détection d'anomalies**, se centre alors sur les capteurs les plus fiables, sans qu'il faille explicitement exclure les capteurs défaillants : ceux-ci s'isolent **d'eux-mêmes** car leurs liens  $\omega_{i,j}$  deviennent insignifiants. Si, par la suite, un capteur perturbé se **rétablit** (ou se recale), les pondérations se remettent à monter, lui permettant de **réintégrer** le cluster d'intérêt. Ce **mécanisme** procure une **robustesse** naturelle face aux dérives ponctuelles de certains capteurs.

Dans un autre contexte, celui d'une **base utilisateur** (profilage, recommandation), il est courant que les **profils** ne renseignent pas l'intégralité des informations attendues. En pratique, un **pipeline** neuronal classique suppose souvent une **imputation** des champs manquants ou l'exclusion des données incomplètes. Le **DSL**, au contraire, autorise l'insertion d'entités représentant « Utilisateur U1 » avec un vecteur  $\mathbf{x}_{U1}$  partiel, ainsi que d'entités « Contenu C1, C2 » définies par divers attributs. La **synergie**  $S(\mathcal{E}_{U1}, \mathcal{E}_{C2})$  tient compte des attributs effectivement présents. Si, même partiellement, l'utilisateur U1 et un contenu C2 trouvent un **recoupement** significatif, la pondération  $\omega_{U1,C2}$  s'accroît sans imposer que tous les champs soient renseignés. Le **réseau** s'**auto-structure** donc malgré l'incomplétude, sans avoir à multiplier les procédures d'imputation ou à écarter systématiquement les exemples incomplets. Cela permet de **préserver** une large portion d'information disponible et d'exploiter la **valeur ajoutée** dès qu'elle se présente.

#### 1.5.3.5. Comment le DSL Surpasse la Rigidité des Réseaux Profonds

Les **réseaux neuronaux profonds** présentent une sensibilité notable aux **incohérences** ou aux cas situés hors de la distribution rencontrée lors de l'entraînement. Si, par exemple, un flux de capteurs n'a jamais inclus des observations particulièrement bruitées, le **modèle** peine à gérer ces perturbations lorsqu'elles apparaissent. En revanche, le **Deep Synergy Learning (DSL)** se caractérise par une **auto-régulation** : lorsqu'une entité  $\mathcal{E}_k$  se montre trop incertaine (synergie très basse ou négative avec d'autres entités), les pondérations  $\omega_{k,j}(t)$  s'amoincrissent, et la contribution de  $\mathcal{E}_k$  se trouve mécaniquement **limitée** ou mise à l'écart, au moins de manière temporaire. Le **SCN** continue malgré tout de fonctionner avec les entités restantes, potentiellement plus fiables.

Dans bien des cas classiques, on recourt à un **modèle d'imputation** (pour combler les données manquantes) ou à des heuristiques rigides visant à traiter le **bruit**. Dans le **DSL**, à l'inverse, chaque entité **évalue** localement la **synergie** possible, sans qu'un algorithme unique de « remplissage » s'impose. Cette démarche revient à laisser chaque **liaison**  $\omega_{k,j}$  apprendre la **compatibilité** ou l'**incompatibilité** entre  $\mathcal{E}_k$  et  $\mathcal{E}_j$ , de sorte que, lorsque le bruit ou l'incohérence survient, la pondération se détériore progressivement, isolant l'entité problématique du **cluster** principal.

Si la **qualité** des données associées à une entité fluctue dans le temps — qu'il s'agisse d'un capteur dont la précision varie ou d'une source intermittente de bruit — la **dynamique** d'auto-organisation gère cette évolution. Les **liaisons**  $\omega_{k,j}$  s'étirent ou se resserrent selon la **synergie** présente, et aucune **réinitialisation** globale ni refonte de l'**architecture** n'est requise. Le réseau d'entités conserve sa topologie, ajustant simplement les **connexions** nécessaires. Cette **flexibilité** évite l'extrême rigidité à laquelle aboutit souvent un réseau profond traditionnel, où l'adaptation à de nouveaux modes de bruit ou d'incomplétude implique souvent un **réentraînement** sur un large jeu de données.

#### Conclusion

La **flexibilité** du DSL face aux **données incomplètes ou bruitées** repose principalement sur :

- L'**auto-organisation**, qui favorise les connexions fructueuses et exclut temporairement les sources peu fiables,

- Le **renforcement local**, où les entités évaluent directement leur synergie avec celles qui détiennent des informations compatibles,
- L'**adaptation continue**, permettant à un capteur rétabli ou à des données initialement partielles de reprendre de la valeur au sein du réseau.

Ce fonctionnement marque un écart fondamental par rapport à la plupart des **réseaux neuronaux profonds**, qui requièrent souvent une **qualité de données** homogène ou un **prétraitement** contraignant. Dans le DSL, la structure est assez **souple** pour intégrer ou ignorer (transitoirement) les entités déficientes, garantissant ainsi une forme de **robustesse** intrinsèque. Après avoir vu (1.5.2) comment le DSL absorbe naturellement la **multi-modalité**, et (1.5.3) comment il gère des données incomplètes ou bruitées, on peut poursuivre (1.5.4) sur son **potentiel d'auto-évolution** et d'adaptation continue, approfondissant encore la dimension dynamique du réseau.

#### 1.5.4. Potentiel d'Auto-Évolution et d'Adaptation Continue

Un des aspects les plus novateurs du **Deep Synergy Learning (DSL)** concerne sa capacité à **évoluer** au fil du temps, sans nécessairement repasser par une phase d'entraînement globale et figée. Plutôt que de geler l'architecture après avoir ajusté quelques poids, le DSL propose une dynamique **en continu** où la structure interne (liens synergiques, clusters, etc.) se reconfigure régulièrement. Cette **auto-évolution** rend le réseau apte à faire face à des changements de distribution (domain shift), à l'apparition de nouvelles entités (nouvelles sources de données) ou à la nécessité d'exploiter un flux continu de données (*streaming data*). Dans cette section (1.5.4), nous examinons :

19. Les principes d'**auto-évolution** du DSL,
20. Comment cette dynamique repose sur l'**adaptation continue** des pondérations (déjà décrite en partie en 1.4.5),
21. Les bénéfices en termes de **Lifelong Learning** (apprentissage tout au long de la vie) et de **plasticité** comparable à des systèmes vivants.

##### 1.5.4.1. L'Auto-Organisation comme Moteur d'Évolution

Dans le **Deep Synergy Learning (DSL)**, la **pondération**  $\omega_{i,j}(t)$  reliant deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  suit la règle de mise à jour décrite en section 1.4.5. La pondération se met à jour selon

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

où  $S(\mathcal{E}_i, \mathcal{E}_j)$  correspond à la **synergie** (distance, similarité, co-information, etc.). Si cette **synergie** demeure suffisamment **positive**, la liaison  $\omega_{i,j}$  se **renforce** ; dans le cas contraire, elle **décroît** et peut passer en deçà d'un **seuil** (voir  $\omega_{\min}$  en section 1.4.5). Un lien sous ce seuil est généralement jugé **inexistant**, reflétant un **désintérêt** ou une **incohérence** mutuelle.

Ce **principe d'évolution dynamique** des connexions fait émerger des **clusters**. Deux entités qui constatent un **gain mutuel** se **rapprochent**, et des sous-groupes peuvent dès lors **fusionner** si la synergie inter-clusters croît au fil des itérations. À l'inverse, un **cluster** se **scinde** quand des sous-groupes décelant davantage de synergie interne choisissent de s'éloigner du groupe initial, leurs connexions inter-groupes  $\omega$  s'abaissant peu à peu faute de bénéfice collectif.

Ce **mécanisme** confère au **DSL** une **dimension** "vivante" : le **Synergistic Connection Network** évolue en continu, s'**adapte** à l'arrivée de nouvelles données ou à la modification d'une source d'information, tout en **préservant** les clusters confirmés et en **permettant** l'émergence de combinaisons inédites. À aucun moment il n'est obligatoire de recommencer un **entraînement** exhaustif du réseau : la **dynamique** locale des pondérations  $\{\omega_{i,j}\}$  assure la **réorganisation** progressive, mettant en place ou dissolvant les **liaisons** selon l'opportunité détectée dans les données.

##### 1.5.4.2. Lifelong Learning et Intégration de Nouvelles Entités

Dans de nombreux environnements dynamiques (robotique, systèmes d'information évolutifs, etc.), il est courant de rencontrer des **données** ou des **entités** inédites au fil du temps. Les réseaux neuronaux profonds classiques se heurtent alors à diverses difficultés : l'ajout d'une nouvelle fonctionnalité peut réclamer la création d'un **module** annexe et un **réentraînement**, parfois complet ou via un **fine-tuning**, avec le risque de compromettre l'équilibre établi. Il arrive aussi que l'on doive réorganiser l'architecture en profondeur, ce qui affecte la stabilité du modèle.

Le **Deep Synergy Learning (DSL)** propose une alternative plus organique. Pour incorporer une **entité**  $\mathcal{E}_{\text{nouvelle}}$  correspondant à un nouveau flux de données, on se contente de l'**introduire** dans le graphe existant. Les pondérations la reliant à d'autres entités  $\{\omega_{\text{nouvelle},i}\}$  s'ajustent

alors en fonction de la synergie détectée. Si une **coopération** apparaît avantageuse, ces liens augmentent et conduisent à l'**insertion** naturelle de l'entité dans un cluster. Il n'est donc pas nécessaire de reconfigurer toute l'architecture : seules les connexions pertinentes se consolident.

Ce mécanisme autorise une **évolution progressive**. Les clusters formés antérieurement conservent leurs **liaisons** tant qu'ils procurent un **gain** : leurs synergies internes demeurent stables. Le réseau n'est pas obligé de sacrifier ses acquis pour introduire de nouvelles entités. Ainsi, on **atténue** le phénomène de "catastrophic forgetting" caractéristique des modèles neuronaux, lesquels risquent d'effacer de vieilles connaissances lorsqu'on leur enseigne de nouvelles tâches. Le DSL, lui, ne **remplace** pas brutalement un jeu de poids par un autre : il **réorganise** plutôt les connexions de manière **locally adaptive**, maintenant la pertinence des anciens clusters et accueillant dans le graphe toute entité inédite jugée utile.

#### 1.5.4.3. Adaptation Continue aux Changements de Distribution

Dans un **réseau** neuronal traditionnel, lorsqu'apparaît un **changement de distribution** dans les données (par exemple l'évolution des conditions de capteurs ou l'apparition de nouvelles caractéristiques), il est souvent indispensable de **réentraîner** ou de **fine-tuner** le modèle, ce qui peut provoquer un ajustement délicat, risqué pour la stabilité ou la précision déjà acquises. En **Deep Synergy Learning (DSL)**, la situation se gère plus soupagement grâce à la **dynamique** auto-organisée qui régit la formation et la déliaison des connexions.

Au fil du temps, la **synergie** entre entités se réévalue en continu. Si un groupe de capteurs cesse de fournir une information fiable, les pondérations associées baissent (faible synergie), déconnectant progressivement l'entité devenue obsolète de la structure ; à l'inverse, si de nouvelles variables ou de nouvelles relations s'avèrent profitables, elles consolident leurs liaisons. Le réseau se remodèle donc de manière **locale**, sans nécessiter la rétropropagation globale ni un réapprentissage exhaustif.

Cette **localité** de la mise à jour, où chaque entité s'occupe prioritairement de ses **liens directs** (et s'ajuste indirectement via les liaisons d'autres entités), confère au **SCN** une forme de **dynamique distribuée**. Les nœuds décident eux-mêmes de renforcer ou d'affaiblir leurs coopérations, suivant l'intérêt (synergie) rencontré, sans orchestration centrale. Le système **s'adapte** ainsi de manière fluide, maintenant une relative **stabilité** pour les groupes demeurant utiles, tout en acceptant l'insertion ou la réduction de connexions pour refléter l'évolution réelle de la **distribution** des données.

#### 1.5.4.4. Comparaison avec l'Architecture Figée d'un Réseau Profond

Dans la majorité des **réseaux neuronaux** classiques (CNN, RNN, MLP, Transformer), l'entraînement s'effectue via **rétropropagation** sur un **ensemble de données** donné a priori, aboutissant à un **modèle** fini. Lorsque la distribution des données évolue ou qu'une nouvelle tâche survient, on se retrouve souvent contraint de réentraîner ou de fine-tuner l'architecture, avec un **risque** d'oubli des acquis précédents (catastrophic forgetting) et un **coût** en calcul potentiellement élevé. Cette rigidité résulte du fait que les **couches** et la **connectivité** du réseau demeurent invariables : les couches convolutionnelles ou self-attention restent fixes, et l'on n'a pas de mécanisme natif pour ajouter ou ôter des neurones ou des flux de données.

Le **Deep Synergy Learning (DSL)** propose une philosophie différente. La structure du **Synergistic Connection Network** est en **évolution continue**. Les entités  $\mathcal{E}_i$  peuvent mettre à jour leurs **représentations** internes  $\mathbf{x}_i$ ,  $\mathbf{s}_i$  et les **liaisons**  $\omega_{i,j}$  s'ajustent (ou se dissolvent) selon le niveau de **synergie**. On peut donc introduire une **nouvelle entité** lorsqu'un flux de données inédits émerge, ou en ôter une qui ne sert plus, sans devoir **réentraîner** le système dans son intégralité. L'**apprentissage** n'est pas un processus qui se clôt : le **réseau** conserve un état de plasticité lui permettant de **s'adapter** en continu aux transformations de l'environnement. Cette approche relève d'un véritable **lifelong learning**, affectant aussi bien les **paramètres** que la **topologie** du SCN. C'est précisément cette capacité de reconfiguration qui différencie la souplesse du **DSL** de la rigidité d'un réseau profond figé, et qui évite de longues phases de réapprentissage global ou de coûteux correctifs pour incorporer de nouveaux concepts ou signaux.

#### 1.5.4.5. Exemples d'Application Pratique

Il existe plusieurs scénarios où l'**approche** de **Deep Synergy Learning (DSL)**, avec son **réseau** auto-organisé et ses liens **adaptatifs**, se révèle particulièrement puissante pour faciliter un **apprentissage continu**. On peut citer :

Dans un contexte de **robotique autonome**, l'environnement et les capteurs peuvent évoluer : la luminosité change, des obstacles inédits surviennent ou un nouveau type de capteur (caméra additionnelle, laser de plus longue portée) est introduit. Au sein d'un **SCN**, chaque capteur ou module de traitement est représenté par une **entité** ; s'il devient peu fiable (par exemple à cause du bruit), ses connexions  $\omega$  s'affaiblissent naturellement, ce qui réduit son influence. Si un nouveau dispositif émerge, il suffit de l'ajouter comme nouvelle entité  $\mathcal{E}_{\text{nouvelle}}$ , et la **synergie** qu'il partage avec d'autres modules s'évaluera de façon autonome. Sans reconstruire l'architecture de zéro, le **DSL** réorganise les liaisons  $\{\omega_{i,j}\}$  et **réalloue** l'importance de chaque capteur selon sa **fiabilité** ou sa pertinence pour la tâche courante.

Dans un **système de recommandation évolutif**, on accueille sans cesse de nouveaux utilisateurs et de nouveaux items (films, livres, produits...). Les méthodes classiques (collaborative filtering, par exemple) requièrent souvent un **réapprentissage** complet ou un



“retrouveau” à intervalles réguliers. Un **DSL**, quant à lui, peut “**brancher**” toute entité  $\mathcal{E}_{\text{new}}$  (nouvel utilisateur ou nouvel objet) au **réseau**, laissant la dynamique de **synergie** ajuster les connexions  $\omega_{\text{new}, \dots}$ . Les liens vers des utilisateurs ou items similaires se verront renforcés au fil des interactions, formant un **cluster** autour de la nouvelle entité. Le coût de recalcul demeure local, et le **réseau** évolue en continu, sans nécessiter un réentraînement global.

Dans l’**analyse contextuelle** (par exemple l’observation de flux de données urbaines ou sociales), la **distribution** peut changer selon l’heure de la journée, la saison, ou les événements. Un réseau traditionnel devrait incorporer différents “régimes” ou recalculer ses poids quand on bascule de “jour” à “nuit” ou d’“hiver” à “été”. Dans un **SCN**, les **clusters** pertinents se forment pour un contexte donné, puis les pondérations  $\omega$  se réactualisent quand le contexte se modifie. Certains sous-groupes se dissolvent si la synergie n’est plus présente, d’autres réapparaissent, et le **DSL** peut ainsi commuter de manière fluide d’un **cluster** à l’autre en fonction de la situation, sans devoir se relancer dans un **apprentissage** lourd.

## Conclusion

Le **potentiel d’auto-évolution** et d’**adaptation continue** du DSL le dote d’une **plasticité** rarement vue dans les architectures d’apprentissage classiques. Son **paradigme** :

- **Ne gèle pas** les liens après entraînement,
- **Accepte** l’arrivée de nouvelles entités ou la disparition d’entités obsolètes,
- **Réajuste** la structure pour épouser l’évolution des données ou du contexte,
- **Garantit** la conservation de l’existant (évite l’oubli brutal) s’il reste utile.

Cette **flexibilité** soutient une vision “**lifelong learning**” où le réseau se maintient en **auto-organisation** permanente, capable d’absorber et d’exploiter des changements de distribution ou des sources inédites. On verra, dans les prochaines sections (1.5.5 à 1.5.7), d’autres arguments majeurs pour une approche synergique (diminution de la supervision, représentations plus riches, intégration de dimensions symboliques).

### 1.5.5. Réduction de la Dépendance à la Supervision Humaine

Au-delà de la multi-modalité (1.5.2), de la gestion des données incomplètes (1.5.3) et de l’adaptation continue (1.5.4), un autre avantage essentiel du **Deep Synergy Learning (DSL)** réside dans sa **relative autonomie** vis-à-vis d’une supervision humaine intensive. Les méthodes traditionnelles de l’IA, notamment les réseaux neuronaux profonds, reposent souvent sur l’accès à d’importants **jeux de données labellisés**, et requièrent un entraînement supervisé (ou semi-supervisé) afin d’ajuster des poids internes. Or :

- Préparer ces labels ou configurations peut être très **coûteux** et **chronophage**,
- L’absence de labels, ou la nécessité d’un apprentissage plus **auto-dirigé**, se fait de plus en plus ressentir dans des contextes réels (big data non étiqueté, environnements inconnus, etc.).

Le **DSL**, du fait qu’il encourage des **relations auto-organisées** entre entités, apporte des mécanismes intrinsèques pouvant réduire la besoin d’étiquettes externes. Cette section (1.5.5) explique :

22. Comment le DSL peut s’**auto-structurer** en s’appuyant sur la synergie plutôt que sur un label,
23. En quoi cela diminue la **dépendance** à la supervision,
24. Quelles **implications** pour l’IA autonome ou non supervisée.

#### 1.5.5.1. L’Auto-Organisation sans Label

Dans les paradigmes d’**apprentissage supervisé**, on dispose en général pour chaque donnée  $\mathbf{x}$  d’un **label**  $y$  (classe, valeur numérique à prédire), et l’entraînement consiste à **minimiser** une fonction de coût  $\mathcal{L}(f_{\theta}(\mathbf{x}), y)$ . On **ajuste** alors les poids du réseau neuronal pour améliorer la correspondance entre l’entrée  $\mathbf{x}$  et la sortie souhaitée  $y$ .

Dans le **Deep Synergy Learning (DSL)**, la dynamique s’appuie sur la **synergie** entre paires d’entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . On définit une quantité  $S(\mathcal{E}_i, \mathcal{E}_j)$  qui mesure le **gain** (ou la complémentarité) qu’elles retirent de leur coopération. Cette fonction de synergie peut reposer sur une **distance** ou une **similarité**, sur la **co-information** en théorie de l’information, ou encore sur tout **critère** mesurant l’utilité mutuelle. Dans

un **réseau** purement non supervisé, on ne dispose d’aucune annotation : la **structure** émerge alors de la manière dont les entités s’**agrègent** ou se **séparent** en fonction de leur synergie locale. Un **objectif** global, le cas échéant, peut exister si un signal partiel ou une contrainte externe oriente la cohérence, mais il n’est pas obligatoire. L’auto-organisation résulte de la **conjonction** des ajustements locaux, sans référence à un label  $y$ .

Cette logique se reflète dans la **formation** de clusters (section 1.4.3). Les entités se regroupent par **renforcement** de leurs connexions si elles détectent un **gain**, et se dissocient sinon. Il n’est pas nécessaire de leur **imposer** des étiquettes comme “classe C1” ou “classe C2” : elles se rassemblent d’elles-mêmes, **non supervisées**, autour de la synergie perçue. Le processus aboutit à un **clustering** naturel, que l’on peut ensuite interpréter comme un groupement de similarités ou de dépendances, et qui peut se révéler **faiblement supervisé** si l’on dispose, en parallèle, de quelques **signaux** ou **contraintes** externes. Dans ce cadre, le **DSL** encourage un **apprentissage local** des coopérations, structurant le **Synergistic Connection Network** sans recourir à un label global pour chaque exemple.

#### 1.5.5.2. Quand un Signal de Supervision Existe...

Le **Deep Synergy Learning (DSL)** n’exclut pas l’éventualité de **labels** ou d’**objectifs** supervisés. Au contraire, il est tout à fait envisageable de disposer d’un **objectif** supervisé pour prédire un label  $y$ , tout en conservant le mécanisme d’**auto-organisation** qui ajuste les liens  $\omega_{i,j}$  en fonction de la synergie. Dans ce scénario, le **DSL** cohabite avec un algorithme de rétropropagation ou d’autre forme de supervision, mais n’est pas entièrement asservi à cet unique signal étiqueté.

Concrètement, on peut imaginer qu’un **objectif** tel que la minimisation d’une fonction de perte  $\mathcal{L}(f_\theta(\mathbf{x}), y)$  se déroule en parallèle de la **dynamique** locale guidée par la synergie. Les **liaisons**  $\omega_{i,j}(t)$  poursuivent la règle adaptative (section 1.4.5), tandis que le **label**  $y$  n’intervient que pour évaluer la cohérence globale ou orienter certains choix de structure. Une partie du réseau peut ainsi se spécialiser dans la tâche supervisée, la **synergie** jouant un rôle déterminant dans l’**organisation** des entités restantes.

Cette architecture présente un **double** bénéfice. D’abord, l’**auto-organisation** ne dépend pas nécessairement du label  $y$ , ce qui autorise un fonctionnement **semi-supervisé** ou même **non supervisé** : la plupart des liaisons  $\omega_{i,j}$  se mettent en place en fonction de la **coopération** intrinsèque, et l’objectif supervisé n’est mobilisé qu’en appoint. Ensuite, si le nombre de labels est **limité**, l’apprentissage reste largement alimenté par la **synergie** interne, évitant la nécessité de disposer d’un jeu de données richement annoté. Le **réseau** peut ainsi, en un premier temps, **structurer** ses entités par pure auto-organisation, puis tirer parti de quelques **labels** additionnels pour affiner la prédiction d’une variable ou la détection d’anomalies.

L’important est que les **labels**, quand ils existent, ne contrôlent pas toutes les **liaisons** du réseau. Seule une fraction des entités ou des connexions peut être orientée par la performance supervisée, tandis que les **clusters** naissent en grande partie de la **synergie** (sections 1.4.3 et 1.4.4). Le réseau se montre donc apte à **apprendre** et **s’adapter** via son auto-organisation, puis à **exploiter** un signal supervisé pour ajuster un module final ou un sous-ensemble d’entités, évitant ainsi la contrainte d’une rétropropagation exhaustive sur toute l’architecture.

#### 1.5.5.3. Exemple : Découverte Spontanée de Catégories

Pour illustrer la manière dont un **Deep Synergy Learning (DSL)** peut structurer les données **sans** s’appuyer sur des labels explicites, on peut considérer un ensemble de **documents** non annotés. Chaque document est alors représenté par un **embedding** ou un **bag-of-words**. Dans le **Synergistic Connection Network**, on introduit autant d’**entités**  $\mathcal{E}_{\text{doc}}$  que de documents, en laissant le mécanisme de **synergie** déterminer comment ces entités se relient.

Les entités commencent par **évaluer** leur proximité ou leur complémentarité : des vecteurs  $\mathbf{x}_{\text{doc}}$  partageant des thèmes (mots récurrents, vocabulaire semblable, distributions de topics) peuvent afficher une **synergie** notable. Les pondérations  $\omega_{i,j}$  entre entités se renforcent si la **similarité** est jugée élevée, ou si la co-information (section 1.4.4) révèle un **gain** dans la mise en commun de leurs contenus. Progressivement, un ou plusieurs **clusters** apparaissent, chaque cluster rassemblant des documents sur des sujets proches, sans qu’on ait besoin de spécifier de labels ou de classes. Ainsi, certains regroupements peuvent porter sur la **politique**, d’autres sur le **sport**, d’autres encore sur la **santé**, mais rien n’impose que le DSL les nomme ou les identifie explicitement.

Si, plus tard, un **expert** fournit quelques **labels** indiquant, par exemple, qu’un certain cluster correspond au domaine sportif, on n’a pas besoin d’annoter l’**intégralité** des documents. Le **DSL** s’était déjà **auto-organisé** en clusters cohérents : il suffit de “**taguer**” postérieurement la communauté d’entités concernée. Ce procédé montre comment un système **non supervisé** ou **faiblement supervisé** peut accomplir une **structuration** fine des données, sur laquelle un label partiel ne fait qu’ajouter un niveau de **description** plus explicite. Les sections antérieures (1.4.3, 1.5.5.1 et 1.5.5.2) soulignent ainsi la flexibilité du DSL pour fonctionner dans des contextes sans labels, tout en permettant une intégration de signaux supervisés quand ils existent.



#### 1.5.5.4. Vers une IA plus Autonome

Les **interactions indirectes** (discutées en section 1.4.6) font en sorte qu’une entité peut être **influencée** par une autre, même en l’absence de lien direct, via des chemins de coopération au sein du **Synergistic Connection Network**. Ce mécanisme enrichit le potentiel d’**auto-découverte** de catégories ou de concepts de manière **non supervisée**, puisque l’information peut circuler librement à travers divers chemins, et les **clusters** émergent en valorisant l’intégralité des signaux disponibles.

Dans une perspective de **IA** plus générale, appelée à affronter des **domaines inconnus** et des **contextes** évolutifs, la **réduction** du besoin de supervision revêt un caractère décisif. Le **Deep Synergy Learning (DSL)**, en misant sur un **apprentissage local et continu**, se montre apte à fonctionner avec des annotations partielles ou inexistantes, tout en conservant la capacité de **structurer** et **organiser** les entités. Il s’agit d’une étape vers une IA plus autonome, qui, sans nécessité d’un étiquetage exhaustif ou de guides fortement supervisés, s’adapte aux environnements **faiblement annotés** (voire non annotés) et continue à explorer la structure interne des données pour faire émerger **concepts** et **catégories** spontanément.

#### 1.5.5.5. Limites et Solutions Partielles

Le **Deep Synergy Learning (DSL)** ne s’appuie pas massivement sur les **labels**, ce qui lui confère une grande liberté dans la formation des **clusters** et dans la découverte de structures internes. Néanmoins, un **petit** signal externe, issu par exemple d’une **semi-supervision**, peut être requis lorsque l’on souhaite orienter la **dynamique** vers des objectifs spécifiques. Sans cette impulsion, le réseau se contente de former des groupes qui font sens pour les données elles-mêmes, sans garantir qu’ils correspondent aux besoins d’une application particulière. Il peut arriver que les clusters, très cohérents sous un angle statistique, ne concordent pas avec la segmentation ou les catégories souhaitées par l’utilisateur.

L’**auto-organisation** peut en effet aboutir à des **partitions** intrinsèquement pertinentes, mais qui ne recouvrent pas exactement les **objectifs** fixés en pratique. Pour rectifier ce phénomène, il est possible d’introduire un **minimum** de **contraintes** ou de **labels** annotés, lesquels jouent le rôle de **pénalités** ou de **feedback** dans la mise à jour des liaisons  $\{\omega_{i,j}(t)\}$ . De la sorte, le DSL ne se limite plus à la seule synergie intrinsèque, mais tient également compte de la finalité appliquée (une tâche métier, une classification imposée), ce qui oriente plus étroitement la **constitution** des clusters.

La **complexité** du calcul de la synergie, surtout pour des données brutes très dimensionnelles, constitue un autre défi. Dans certains cas, on envisage des mesures plus poussées telles que la **co-information** ou la **Partial Information Decomposition** (sections 1.4.4 et 1.4.7), dont l’estimation peut s’avérer lourde. Des **estimateurs** non paramétriques requièrent souvent un échantillonnage important et des ressources conséquentes. Il est alors crucial de se doter de **stratégies** d’approximation ou d’**échantillonnage** (sampling parcimonieux, heuristiques de calcul) pour maintenir la faisabilité sur des volumes de données massifs (big data). Cette solution, certes partielle, demeure incontournable si l’on veut étendre le **DSL** à des scénarios industriels exigeants, tout en préservant un **coût** de calcul acceptable.

### Conclusion

Le **Deep Synergy Learning** offre un **cadre** qui diminue la **dépendance** à la **supervision humaine** en se basant principalement sur :

- Des **fonctions de synergie** (distance, similarité, information mutuelle, etc.) pour guider la structuration,
- Une **auto-organisation** où les entités coopèrent ou se séparent localement,
- La possibilité d’un **apprentissage** non supervisé, ou faiblement supervisé, via la croissance / décroissance des pondérations  $\omega_{i,j}$ .

Cette **indépendance** accrue vis-à-vis d’un étiquetage exhaustif, couplée à la **plasticité** et à l’**adaptation continue** (1.5.4), fait du DSL un **candidat** privilégié pour les systèmes d’IA évolutifs, explorant des environnements peu annotés ou devant se reconfigurer en permanence. Par la suite (1.5.6, 1.5.7), nous verrons comment cette approche peut encore être renforcée pour créer des représentations **riches** et éventuellement **cognitives**, en incorporant des idées de **conceptualisation** ou de **symbolique**.

#### 1.5.6. Création de Représentations Riches et plus Interprétables

En plus de la **réduction de la dépendance à la supervision** (1.5.5), le **Deep Synergy Learning (DSL)** présente un avantage notable pour la **création de représentations** à la fois **riches** et **interprétables**. Contrairement aux réseaux neuronaux profonds classiques, qui produisent souvent des représentations “boîte noire” difficilement explicables, le DSL mise sur l’**auto-organisation** et la **coopération** entre entités d’information, ce qui peut aboutir à des structures internes (clusters, macro-clusters, liens synergiques) plus compréhensibles pour un humain. Cette section (1.5.6) met en évidence :

25. **Pourquoi** la synergie favorise l’émergence de **représentations complexes**,

26. Comment le principe de **clusters** ou de **macro-clusters** rend l'organisation plus **lisible**,
27. En quoi le DSL peut faciliter l'**interprétabilité** par rapport aux architectures profondes traditionnelles.

#### 1.5.6.1. Logique “par Entités” plutôt que “en Couches Opaques”

Dans les **réseaux neuronaux profonds** (CNN, RNN, Transformers...), la **représentation** interne s'étage à travers plusieurs **couches** successives de transformations non linéaires. Malgré certaines avancées, telles que la visualisation de **filtres** dans un CNN ou l'analyse de **matrices d'attention** dans un Transformer, comprendre comment le réseau aboutit à sa décision globale demeure un défi majeur. Les poids ne sont pas naturellement groupés sous forme de blocs “légibles” et, hors du cadre spécifique d'un neurone ou d'une couche, il est difficile de rattacher un sous-ensemble de paramètres à un **concept** clairement identifiable. Les techniques d'**interprétabilité** (comme Grad-CAM ou LIME) tentent de mettre en évidence l'influence de certains pixels ou de certaines dimensions, mais elles proposent souvent des **heuristiques** ne fournissant pas nécessairement une vue d'ensemble de la logique interne.

Le **Deep Synergy Learning (DSL)** adopte une approche sensiblement différente en structurant l'information autour d'**entités** (nœuds du **Synergistic Connection Network**). Chaque entité peut représenter un **flux sensoriel** ou un **descripteur** plus abstrait (embedding, concept partiel). Les **liaisons** entre entités indiquent la **synergie** détectée, c'est-à-dire la plus-value mutuelle de leur coopération. Ainsi, la représentation n'est plus seulement un pipeline linéaire ou une série de couches opaques : elle se formalise comme un **graphe** de modules où l'on identifie plus aisément les fragments d'information et où l'on constate quelles entités collaborent avec quelles autres. Les **clusters** qui émergent se trouvent alors naturellement interprétables : un sous-groupe peut regrouper la “détection de visages” (vision), l’“audio de voix” (son) et un “module de reconnaissance textuelle” (parole transformée en mots-clés), offrant un tableau beaucoup plus modulaire et segmenté qu'un simple empilement de couches neuronales classiques.

Cette logique “par entités” rend donc la représentation moins opaque : on sait, au sein du **SCN**, quelles entités existent, quelles sont leurs fonctions ou données principales et comment elles s'**agencent** via les pondérations de synergie. Le **DSL** propose ainsi un degré d'**explicabilité** supplémentaire, puisque la structure d'ensemble apparaît comme un **réseau** plus proche d'une **cartographie** du flux d'information, plutôt qu'un empilement difficile à démêler d'unités et de poids dispersés.

#### 1.5.6.2. Clusters et Macro-Clusters comme Briques de Sens

Dans le **Deep Synergy Learning (DSL)**, lorsqu'un **cluster** d'entités se forme (section 1.4.3), il peut refléter un **concept** ou un **thème** commun aux données. Imaginons un **cluster** multimodal  $\{\mathcal{E}_{\text{image}}, \mathcal{E}_{\text{audio}}, \mathcal{E}_{\text{texte}}\}$  qui se spécialise dans la détection d'un “événement conférence” parce que l'entité image renvoie à une scène d'orateur faisant face à un public, l'entité audio capture un bruit de voix ou d'applaudissements, et l'entité texte détecte des mots comme “bienvenue” ou “question”. La **cohérence** ainsi dégagée à l'intérieur du cluster n'est pas un simple effet de poids numériques : on sait **quelles** entités s'agrègent, **pourquoi** (à travers la synergie évaluée localement), et cela confère à l'ensemble une **lisibilité** accrue.

Du point de vue de l'**explicabilité** ou de l’“Explainable AI”, un **cluster** dans le **Synergistic Connection Network** n'équivaut pas à un bloc opaque de neurones comme dans un réseau profond traditionnel. Il s'agit plutôt d'un **sous-graphe** composé d'entités portant une signification distincte (capteurs visuels, flux audio, modules textuels, ou encore vecteurs symboliques...). Les **liaisons synergiques**  $\omega_{i,j}$  reflètent la force de leur coopération : on peut par exemple examiner la distribution de la co-information ou de la similarité entre chaque paire. Les regroupements (macro-clusters) où plusieurs clusters se **fusionnent** indiquent qu'un ensemble plus large d'entités se renforcent réciproquement dans leur coopération. Dans un **contexte e-commerce**, on pourrait alors voir un macro-cluster associer plusieurs catégories de produits, des groupes d'utilisateurs et des attributs de profil, dessinant un **sous-réseau** particulièrement actif et riche en **synergie**.

Ces **briques** (clusters, macro-clusters) constituent autant de **modules** que l'on peut **interpréter** et **commenter** : chaque entité, de par son lien avec un flux de données ou un vecteur caractéristique particulier, donne un **ancrage** explicite à son rôle. Les liens montrent la proportion de **synergie** interne, et le **DSL** scinde le graphe global en plusieurs **sous-ensembles** correspondant à des **concepts** ou **sujets**. Cette granularité rend la représentation plus modulaire, invitant à un examen plus détaillé de l'organisation du réseau et offrant des pistes de justification plus tangibles que la simple consultation de poids dans une couche neuronale abstraite.

#### 1.5.6.3. Représentations Multiples et Non Linéaires

Le **Deep Synergy Learning (DSL)** ne se borne pas à des représentations linéaires ou statiques. Au contraire, chaque **entité**  $\mathcal{E}_i$  peut conserver une **représentation interne**  $\mathbf{x}_i(t)$  ou  $\mathbf{s}_i(t)$  qui évolue au fil du temps, ou être associée à un petit réseau neuronal local (par exemple un autoencodeur) calculant

$$\mathbf{x}_i(t+1) = F_i(\mathbf{x}_i(t), \{\omega_{i,k}(t)\}, \dots).$$

La **non-linéarité** inhérente à ce type de module (RNN local, MLP, autoencodeur) élargit considérablement l'espace des **représentations** envisageables, comparativement à un simple cadre vectoriel fixe. Les entités peuvent ainsi affiner leurs **features** internes ou leurs **états**  $s_i$ , tout en mettant à jour leurs **liaisons**  $\omega_{i,j}(t)$  d'après la **synergie** nouvellement constatée. Cette approche ne constitue pas un pipeline rigide : chaque entité, considérée comme un bloc fonctionnel autonome, maintient sa propre logique d'évolution, tandis que la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  se réévalue localement.

Si deux entités en viennent à **coïncider** fortement (en partageant des représentations très similaires ou très complémentaires), leurs pondérations  $\omega_{i,j}$  peuvent atteindre un niveau **saturé**, conduisant potentiellement à une **fusion** (au sens de la fusion de clusters, section 1.4.3). Inversement, d'autres entités peuvent demeurer **indépendantes**, préférant garder leurs liaisons minimales. Le résultat global est une **représentation** dite "multi-foyers", où plusieurs **clusters** se spécialisent dans des sous-ensembles de la tâche ou des données. Chacun possède son propre noyau fonctionnel, et ils coopèrent occasionnellement si la **synergie** l'exige. Cette organisation **modulaire** tranche avec un réseau profond monolithique dans lequel toutes les informations finissent entremêlées au sein de couches successives. Elle favorise un **équilibre** entre la **richesse** (via la non-linéarité) et la **lisibilité** (chaque entité reste limitée à un bloc fonctionnel clairement cerné), tout en laissant l'auto-organisation diriger la consolidation ou la séparation des diverses composantes.

#### 1.5.6.4. Comparaison avec l'Explicabilité dans les Réseaux Profonds

Dans les approches classiques d'**explainability** appliquées aux réseaux neuronaux profonds (CNN, Transformers, etc.), il est habituel de recourir à des méthodes "post-hoc", comme la visualisation d'**activation maps**, l'explication locale (LIME, SHAP) ou l'examen des **attention maps** dans un Transformer. Ces techniques fournissent un aperçu de la raison pour laquelle le réseau met en avant tel pixel, telle dimension ou tel token, mais elles ne suppriment pas pour autant le caractère opaque de la structure interne. Un CNN reste un empilement de couches, au sein desquelles la signification des poids demeure largement cachée, et un Transformer recèle une superposition de blocs d'attention dont la lisibilité globale n'est pas toujours aisée.

Le **Deep Synergy Learning (DSL)** offre une **explication intrinsèque** plus directe, car l'information s'organise en entités reliées par des **liaisons**  $\omega_{i,j}$  qui traduisent la synergie entre ces entités. Le graphe obtenu est alors lisible : on distingue les **clusters** et macro-clusters correspondant à des groupes d'entités coopératives, et chaque entité garde une identité claire (modalité, bloc fonctionnel, concept partiel). L'utilisateur peut examiner un cluster  $\mathcal{C}$  pour déterminer quelles entités le composent, quelles caractéristiques portent chacune, comment se distribuent les pondérations  $\omega_{i,j}$ . De là, il est possible de déduire la logique d'assemblage : on voit, par exemple, qu'un sous-graphe associe la détection de panneaux de signalisation à l'enregistrement de la vitesse et au suivi de la trajectoire, ce qui éclaire la raison d'être de ce regroupement. Un tel niveau de transparence demeure rarement accessible dans un réseau profond standard, où l'on ne peut guère rattacher des ensembles de neurones internes à une fonction ou à un concept identifié, si ce n'est par l'ajout ultérieur d'artifices explicatifs. Le DSL rend la structure explicite et modulable, montrant clairement **qui** coopère avec **qui**, et dans quelle mesure, par l'intermédiaire de pondérations synergiques réparties sur l'ensemble du **Synergistic Connection Network**.

#### 1.5.6.5. Exemple d'Application : Représentation Sémantique Évolutive

Dans un **agent conversationnel**, plusieurs **entités** peuvent correspondre à différents volets de l'interaction : ainsi, une entité  $\mathcal{E}_{\text{linguistique}}$  traite la compréhension du langage, une entité  $\mathcal{E}_{\text{contexte}}$  se réfère aux informations contextuelles (heure, lieu, historique de conversation), et une entité  $\mathcal{E}_{\text{profil utilisateur}}$  incarne les préférences ou les caractéristiques de l'utilisateur. Au fil des échanges, les synergies entre ces composantes peuvent croître lorsque, par exemple, le  $\mathcal{E}_{\text{profil}}$  apporte des indices qui concordent avec  $\mathcal{E}_{\text{linguistique}}$ . Le **DSL** favorise alors la formation d'un **cluster**  $\{\mathcal{E}_{\text{ling}}, \mathcal{E}_{\text{contexte}}, \mathcal{E}_{\text{profil}}\}$  dont la **coopération** explicite se traduit par la hausse des pondérations  $\omega_{i,j}$ . Il en résulte un "sous-modèle" sémantique plus clair, articulé autour des thèmes, des intentions ou des préférences récurrentes du locuteur. En examinant la façon dont les liens se sont construits, on peut identifier les **topics** qui émergent et comprendre comment s'est bâtie cette synergie.

Dans un autre registre, l'**analyse d'images documentaires** peut tirer profit de ce principe. Certaines **entités**  $\mathcal{E}_{\text{patch}}$  décrivent les patches ou segments visuels (éléments d'architecture, styles picturaux), alors que d'autres entités regroupent des **mots-clés** extraits de légendes ou de métadonnées, ou encore des **concepts** plus abstraits liés à une époque ou un auteur. Il se crée ainsi un **cluster** combinant, par exemple, un patch visuel P1, un ensemble de mots-clés M2 et une entité conceptuelle C3 qui indique la période médiévale. L'auto-organisation du **Synergistic Connection Network** montre que la **co-information** est élevée : les motifs visuels rappellent l'architecture gothique, les mots-clés renvoient à un style médiéval, et la dimension conceptuelle (C3) représente cet intervalle historique. La **coopération** entre ces entités confère une **lisibilité** accrue : on identifie un regroupement signifiant « document archéologique médiéval » sans avoir dû imposer de règle au préalable. Le **DSL** opère donc comme un **pilier** sémantique dynamique, consolidant les liens lorsqu'il perçoit un bénéfice mutuel et autorisant une **représentation** riche et modulable.

## Conclusion

La **création de représentations riches et plus interprétables** fait partie des **points forts** du DSL, grâce à :

- Son **organisation par entités** (nœuds clairs, porteurs d’une modalité ou d’une fonction),
- Sa **clusterisation** auto-organisée, permettant d’identifier *qui coopère avec qui*,
- La **possibilité** d’observer directement les liaisons  $\omega_{i,j}$  et leurs pondérations,
- Un **niveau de modularité** qui dépasse l’approche “monobloc” des réseaux neuronaux profonds.

De ce fait, le DSL se prête mieux à des besoins de **transparence**, de **compréhension** et de **diagnostic**. Couplée aux atouts précédents (multi-modalité, gestion du bruit, adaptation continue, moindre dépendance à la supervision), cette caractéristique renforce l’idée que l’approche synergique peut offrir une **IA plus agile**, pouvant aboutir à des **configurations** internes plus **accessibles** à l’analyse humaine. Dans la dernière section (1.5.7), nous verrons un autre aspect crucial : l’**intégration** de **dimensions symboliques** ou **cognitives** dans un cadre DSL, promettant une forme encore plus avancée d’**IA unifiée**.

### 1.5.7. Intégration de Dimensions Symboliques ou Cognitives

Les sections précédentes (1.5.1 à 1.5.6) ont souligné diverses forces du **Deep Synergy Learning (DSL)**, telles que la **capacité à gérer la multi-modalité**, à s’**adapter** en continu, ou à **créer** des représentations plus **lisibles**. Un autre point capital, souvent abordé dans l’IA contemporaine, est la possibilité de **mélanger** des composantes **sub-symboliques** (issues des méthodes connexionnistes) et des composantes **symboliques** (notions logiques, sémantiques, règles expertes). Alors que les réseaux neuronaux profonds “classiques” ont généralement peiné à intégrer ces dimensions symboliques, le **DSL**, par sa structure en entités et liens synergiques, peut être un **cadre** propice pour unir symbolique et sub-symbolique.

Cette section (1.5.7) explique :

28. Comment le DSL autorise l’**introduction** d’entités à **portée symbolique** (règles, concepts, modèles cognitifs),
29. Pourquoi cette intégration est **plus fluide** que dans un réseau hiérarchique figé,
30. Les **applications** possibles (raisonnement, inférence, cognition proche de l’humain),
31. Les **défis** mathématiques associés à la cohabitation de ces deux modes de représentation.

#### 1.5.7.1. Aperçu des Approches Symboliques et Connexionnistes en IA

Dans le domaine de l’**IA**, deux grands courants se distinguent historiquement. Le premier, souvent qualifié de **IA symbolique**, s’appuie sur des **représentations logiques** et des **moteurs d’inférence** capables de manipuler des symboles pour résoudre des problèmes. Cette approche offre une **explicabilité** naturelle, car les **règles logiques** sont décrites dans un langage clair et les chaînes de raisonnement peuvent être retracées. Il en découle une forme de **transparence** : on peut savoir précisément pourquoi une conclusion a été tirée. Cependant, ce paradigme symbolique se heurte à plusieurs limites. Il gère mal l’**incertitude** et se révèle peu adapté à l’**apprentissage** automatique à partir de données massives ou bruitées. L’idée d’extraire directement des **motifs** complexes d’un grand ensemble d’observations, sans intervention humaine pour écrire les règles, demeure délicate dans ce cadre.

À l’opposé, les méthodes **connexionnistes**, dont les **réseaux neuronaux** constituent l’archétype, excellent dans l’**apprentissage** à partir de larges volumes de données hétérogènes. Elles reposent sur des **paramètres** réels ajustés par descente de gradient ou par d’autres algorithmes, permettant de capturer des régularités complexes dans l’**espace** des entrées. L’exemple canonique est la classification d’images, où un réseau convolutif identifie des motifs tout en s’ajustant aux écarts et aux bruits des données. Toutefois, ce cadre connexionniste se heurte à une **opacité** interne, souvent résumée par l’expression “boîte noire”. Les poids et activations qui conduisent à la décision sont difficiles à **expliquer**, et la notion de **règles logiques** ou de **concepts** abstraits y est absente dans la formulation de base. Il n’existe pas de mécanisme formel de **raisonnement** symbolique, ni de vérification de **cohérence** par rapport à un ensemble de connaissances explicites.

Dans les sections suivantes, le **Deep Synergy Learning (DSL)** sera envisagé comme un **cadre unificateur** susceptible d’exploiter les forces des deux approches. L’accent sera mis sur la manière de tirer parti de la **plasticité** et de la **puissance** d’apprentissage sub-symbolique tout en intégrant, au sein d’un **Synergistic Connection Network**, des **entités** représentant des **règles** ou **concepts** symboliques. Ainsi, on peut espérer concilier l’**adaptation** continue des réseaux neuronaux avec la **lisibilité** et la **transparence** associées aux logiques symboliques.

### 1.5.7.2. Principes d'une Approche Hybride Symbolique–Connexionniste

Dans la continuité des sections précédentes, et en particulier au regard des principes généraux du **Deep Synergy Learning (DSL)**, on constate que les approches **symboliques** et **sub-symboliques** apparaissent de plus en plus **complémentaires**. Les récentes avancées en **IA** soulignent cette complémentarité : d'un côté, les méthodes sub-symboliques (réseaux neuronaux classiques, **DSL** et paradigmes voisins) démontrent une **puissance d'apprentissage** remarquable lorsqu'il s'agit de traiter de grandes quantités de données, ou d'extraire des **caractéristiques** non triviales dans un **espace** de dimension élevée. De l'autre, les méthodes symboliques (appuyées sur des **logiques**, des **règles** et des **ontologies**) assurent une **rigueur** et une **explicabilité** supérieures, notamment lorsqu'il convient de formuler des raisonnements structurés ou de décrire des **connaissances** de manière **interprétable**.

La question est donc de **concevoir** une architecture **hybride** qui mobilise simultanément la **puissance d'apprentissage** sub-symbolique et la **cohérence symbolique**. Dans le cadre d'un **Synergistic Connection Network (SCN)** tel que défini dans les sections précédentes, il est possible d'**introduire** des entités de différents types. Il est ainsi envisageable de faire cohabiter des **entités**  $\mathcal{E}_{\text{symb}}$  représentant des **concepts logiques**, des **règles formelles**, ou des **faits** structurés, et des **entités**  $\mathcal{E}_{\text{data}}$  se fondant sur des **vecteurs sub-symboliques**  $\mathbf{x}_i \in \mathbb{R}^d$ . On peut imaginer que la représentation symbolique contienne des énoncés tels que « Un véhicule se déplace sur roues » ou « Si la température excède 100\text{°C}, on suspecte une surchauffe », tandis que la partie sub-symbolique gère des **flux de données** (images, séries temporelles, signaux sensoriels) ou des **features** issues de réseaux de neurones.

Dans cette optique, on peut noter  $\mathcal{E}_{\text{symb}}$  pour une **entité symbolique** et  $\mathcal{E}_{\text{data}}$  pour une **entité sub-symbolique**. La **pondération**  $\omega_{\text{symb,data}}(t)$  s'adapte alors par la règle classique du **DSL** :

$$\omega_{\text{symb,data}}(t+1) = \omega_{\text{symb,data}}(t) + \eta [S(\mathcal{E}_{\text{symb}}, \mathcal{E}_{\text{data}}) - \tau \omega_{\text{symb,data}}(t)],$$

où la **synergie**  $S(\mathcal{E}_{\text{symb}}, \mathcal{E}_{\text{data}})$  traduit à quel point la **règle** ou le **concept** véhiculé par  $\mathcal{E}_{\text{symb}}$  s'applique utilement au **contenu** sub-symbolique porté par  $\mathcal{E}_{\text{data}}$ . La valeur de  $S$  peut tenir compte du **taux de succès** lors de la confrontation aux données, ou évaluer la **compatibilité** sémantique entre la règle symbolique et les caractéristiques identifiées dans le **flux sub-symbolique**.

On peut approfondir cette **coopération** à l'aide de **formules** décrivant la mise en correspondance entre un **concept** symbolique et des **features** extraites. Par exemple, on peut écrire :

$$S(\mathcal{E}_{\text{symb}}, \mathcal{E}_{\text{data}}) = \text{score}(\text{matching}(\text{rules}(\mathcal{E}_{\text{symb}}), \mathbf{x}_{\text{data}})),$$

où  $\text{rules}(\mathcal{E}_{\text{symb}})$  dénote l'ensemble des règles ou faits associés à la représentation symbolique, et  $\mathbf{x}_{\text{data}}$  la **représentation vectorielle** de l'entité sub-symbolique. La fonction **matching** évalue la cohérence entre la **logique symbolique** et les **observations** sub-symboliques, et **score** traduit le **gain** ou la **conformité** obtenu.

Dans cette approche **hybride**, il y a un **avantage** notable. Le réseau **auto-organisé** du DSL peut **identifier** quelles **règles** ou **concepts** sont les plus pertinents pour tel **type de données**, sans nécessiter une supervision massive. Les entités symboliques, en s'activant sur certaines conditions, peuvent **expliquer** plus aisément les **décisions** ou **groupements** effectués. La **transparence** s'en trouve améliorée : si la **pondération**  $\omega_{\text{symb,data}}$  monte en régime, cela indique que la règle symbolique s'applique fréquemment et avec succès à la modalité de données considérée.

La **limite** principale repose sur la **cohérence** nécessaire entre la **logique** et les **données**. L'entité symbolique doit être suffisamment bien définie et adaptée à la réalité sub-symbolique ; en cas de décalage, la pondération associée risque de rester faible, rendant la règle peu intégrée. Sur le plan de l'implémentation, il faut également gérer la **complexité** liée à l'activation symbolique, potentiellement coûteuse s'il faut évaluer de nombreuses règles sur un flux volumineux de données.

Dans ce **paradigme**, on constate néanmoins que la **puissance** du DSL est préservée : le réseau peut se reconfigurer en continu, renforçant ou affaiblissant les liens entre entités symboliques et sub-symboliques selon la **synergie** détectée. En parallèle, les entités symboliques aident à **structurer** le raisonnement et à **rendre compte** de la logique sous-jacente, contribuant à l'**explicabilité** du système final. C'est cette double souplesse qui, dans la ligne de ce chapitre (section 1.5.7.2), semble prometteuse pour de nombreuses applications, allant de la **robotique cognitive** à l'**intégration** de vastes **bases de connaissances** et de **flux sensoriels**.

### 1.5.7.3. Mécanismes d'Auto-Organisation Intégrant la Logique

Dans la continuité de l'approche **hybride** symbolique–connexionniste présentée en section 1.5.7.2, il est envisageable d'introduire au sein d'un **Synergistic Connection Network (SCN)** des **règles symboliques** interagissant avec des **entités** sub-symboliques. On peut par exemple considérer une **règle**  $R$  stipulant :

Si la caméra détecte une forme circulaire, alors suspecter un panneau de signalisation.

Parallèlement, on introduit une **entité sub-symbolique**  $\mathcal{E}_{\text{cam}}$  chargée d'analyser des données visuelles. Cette entité exploite typiquement un **modèle** neuronal local pour reconnaître ou segmenter des formes dans un **flux** d'images.

La **pondération**  $\omega_{R, \text{cam}}$  reliant la règle  $R$  à l'entité  $\mathcal{E}_{\text{cam}}$  suit la dynamique du **DSL**. On peut ainsi l'écrire :

$$\omega_{R, \text{cam}}(t + 1) = \omega_{R, \text{cam}}(t) + \eta [S(R, \mathcal{E}_{\text{cam}}) - \tau \omega_{R, \text{cam}}(t)],$$

où

$S(R, \mathcal{E}_{\text{cam}})$  mesure la **synergie** entre la **règle symbolique** et la **détection sub-symbolique** opérée par  $\mathcal{E}_{\text{cam}}$ . Concrètement, si la **forme circulaire** repérée par la caméra coïncide fréquemment avec la proposition « panneau de signalisation », la quantité  $S(R, \mathcal{E}_{\text{cam}})$  prend une **valeur positive**. Le terme  $\eta [S - \tau \omega]$  reste alors **positif**, ce qui **renforce** la liaison  $\omega_{R, \text{cam}}$ . Progressivement, on voit s'**organiser** un **cluster** local  $\{R, \mathcal{E}_{\text{cam}}\}$ , soulignant la coopération entre la **règle symbolique** et la **détection d'images**.

Lorsque survient une **deuxième** entité,  $\mathcal{E}_{\text{limit}}$ , qui incarne un **concept** tel que « panneau de limitation de vitesse », la **coopération** peut encore se prolonger. Si un **module**  $\mathcal{E}_{\text{ocr}}$  découvre régulièrement le texte « 30 km/h » à l'intérieur des formes circulaires détectées, la synergie  $S(\mathcal{E}_{\text{ocr}}, \mathcal{E}_{\text{limit}})$  peut s'avérer **élevée**. La **pondération**  $\omega_{\text{ocr}, \text{limit}}$  croît alors, impliquant l'émergence d'un **sous-réseau** plus vaste :

$$\{R, \mathcal{E}_{\text{cam}}, \mathcal{E}_{\text{ocr}}, \mathcal{E}_{\text{limit}}\}.$$

À l'intérieur de ce **cluster**, les entités **symboliques** (par exemple  $R$  et  $\mathcal{E}_{\text{limit}}$ ) formulent des règles logiques ou des connaissances à caractère explicite. Les **entités**  $\mathcal{E}_{\text{cam}}$  et  $\mathcal{E}_{\text{ocr}}$  se chargent au contraire d'un **traitement** sub-symbolique (détection de formes, reconnaissance du texte « 30 »). La **dynamique** des pondérations, selon la règle linéaire ou d'autres variantes (voir section 1.4.5), veille à **consolider** cet ensemble si les corrélations s'avèrent régulières et profitables.

Dans cette **topologie**, le **raisonnement** reste en grande partie **distribué** et local : il n'y a pas une unique couche hiérarchique, mais une **auto-organisation** en **micro-réseaux** où règles et modules neuronaux s'**associent** dès que leur **synergie** atteint un **niveau** significatif. Les **avantages** de cette intégration résident dans la capacité à **exploiter** la puissance de l'apprentissage sub-symbolique pour reconnaître des patterns visuels complexes, tout en s'appuyant sur la **précision** et la **transparence** qu'offrent les entités symboliques. De surcroît, on peut ainsi relier de multiples règles, chacune s'appliquant à des **entités** sub-symboliques différentes, ce qui permet d'atteindre une **modularité** plus élaborée à l'intérieur du **SCN**.

Les **limites** et **inconvénients** tiennent essentiellement au besoin de concevoir des règles symboliques suffisamment précises pour bien refléter la réalité observée par les flux sub-symboliques. S'il existe un décalage important ou un **bruit** excessif dans la détection, la synergie peut rester faible et empêcher la formation d'un cluster stable. Un autre **défi** réside dans la gestion de la **scalabilité** : si un trop grand nombre de règles ou d'entités sub-symboliques sont présentes, leur mise en correspondance peut devenir coûteuse à calculer, ce qui impose des stratégies de **sparsification** ou de **pruning** (cf. section 1.4.5).

Dans l'ensemble, cette **intégration** de la **logique** dans le **DSL** illustre la possibilité de **fusionner** la robustesse neuronale ou statistique et la **structuration** symbolique. L'**auto-organisation** garantit que seules les entités **règle-module** qui partagent un gain effectif conservent un lien fort, aboutissant à une **cohérence** émergente entre les **concepts** logiques et les **vecteurs** de features dans le **Synergistic Connection Network**.

#### 1.5.7.4. Avantages pour la Cognition et l'Explicabilité

Dans la continuité des sections précédentes, le **Deep Synergy Learning (DSL)** se présente comme un **pont** unifiant deux grandes approches du traitement de l'information. D'une part, la **dimension sub-symbolique** s'attache aux aspects de **perception** et de **features** extraits directement des données (images, sons, textes bruts), héritant de la puissance d'**apprentissage** de type connexionniste. D'autre part, la **dimension symbolique** articule des **concepts**, des **règles** et des **faits** logiques qui ne seraient pas forcément découverts par simple analyse statistique, par exemple : « un véhicule possède quatre roues, un moteur et est conçu pour se déplacer sur la route ».

Lorsqu'un **cluster** (au sens de la section 1.4.3) associe une **règle** formalisée à des **entités perceptives** ancrées dans des vecteurs sub-symboliques, on obtient une **forme d'explication** plus directe de la décision. En effet, plutôt que de faire référence à un poids numérique situé dans une couche profonde du réseau, on peut pointer vers le **signal**  $\mathbf{x}$  (issu de la représentation sub-symbolique) et la **règle**  $R$  (exprimée symboliquement) pour justifier la conclusion. Cette **transparence** s'apparente à une forme d'**explicabilité** (souvent recherchée en IA), où l'on peut réellement indiquer quelles **connaissances** et quelles **observations** ont motivé l'activation finale.

Pour formaliser mathématiquement cette idée, on peut écrire que la **pondération**  $\omega_{R, \text{data}}(t)$  reliant la règle symbolique  $\mathcal{E}_R$  à l'entité sub-symbolique  $\mathcal{E}_{\text{data}}$  évolue selon :

$$\omega_{R, \text{data}}(t + 1) = \omega_{R, \text{data}}(t) + \eta [S(\mathcal{E}_R, \mathcal{E}_{\text{data}}) - \tau \omega_{R, \text{data}}(t)],$$

où  $S(\mathcal{E}_R, \mathcal{E}_{\text{data}})$  mesure l'**adéquation** entre la règle symbolique et les **caractéristiques** perçues dans les données. Une fois cette liaison suffisamment renforcée, on peut interpréter la **décision** ou la **catégorie** apprise en mentionnant explicitement la règle  $R$  et le segment de données  $\mathbf{x}$  qui la **valide**.

Ce mécanisme illustre la notion de **cognition synergique** : les **entités** symboliques **codent** la **sémantique** et les **règles** explicites, tandis que les **entités** sub-symboliques **apprennent** à partir des données brutes, en extrayant des **patterns** ou des **features** utiles. Dans le **Synergistic Connection Network (SCN)**, la **dynamique** de mise à jour des **pondérations** relie directement ces deux pôles. On aboutit



alors à une forme d'**intégration** qui se rapproche de la **cognition humaine**, où des **concepts** explicitement formulés cohabitent avec des **associations perceptives** acquises à travers l'expérience.

En termes d'**avantages**, on dispose d'une **robustesse** caractéristique des méthodes sub-symboliques, puisque le **réseau** peut s'adapter aux données, tout en bénéficiant de la **précision** et de la **lisibilité** qu'offre la **logique symbolique**. Cela se traduit par un **apprentissage** plus puissant combiné à une **explication** plus accessible : la conclusion ne dépend plus seulement d'une somme opaque de poids, mais peut être reliée à un **ensemble** (règle, concept, fait) qui rend compte de sa **signification**. Dans le même paragraphe, on peut souligner que la **cohabitation** de règles symboliques et de flux sub-symboliques exige une **cohérence** de haut niveau. Les règles doivent être appropriées pour décrire les phénomènes observés, et les entités sub-symboliques doivent parvenir à traduire efficacement le **signal** en indices qui **valident** ou **invalident** les **règles**.

Cette **fusion** entre ces deux mondes (symbolique et sub-symbolique) au sein du **DSL** peut ainsi instaurer un **climat** proche de ce qu'on appelle la "**cognition synergique**" : à la manière de l'esprit humain, le système mixe des **concepts** explicites et des **représentations** perceptives implicites, avant de les **coordonner** dynamiquement pour décider de l'interprétation finale.

### 1.5.7.5. Défis Mathématiques et Implémentation

Dans la poursuite des principes décrits en sections 1.5.7.2 à 1.5.7.4, se pose la question de la **représentation** mathématique des **règles** symboliques à l'intérieur du **Deep Synergy Learning (DSL)** et, plus généralement, les difficultés techniques qu'implique l'introduction de composantes **logiques**. Un premier enjeu consiste à décider comment **encoder** une règle  $\mathcal{E}_{\text{rule}}$ . Certains travaux préconisent d'associer à chaque règle un **vecteur** ou un **arbre** logique compact, ce qui permet d'attribuer à  $\mathcal{E}_{\text{rule}}$  un **nœud** du réseau muni de **paramètres**  $\theta_{\text{rule}}$ . On peut alors définir la **synergie**  $S(\mathcal{E}_{\text{rule}}, \mathcal{E}_{\text{feature}})$  de façon à intégrer à la fois la **structure symbolique** (par exemple, un parse-tree) et la **partie numérique** (par exemple, un embedding vectoriel ou un ensemble de features  $\mathbf{x}_{\text{data}}$ ).

Il est possible d'écrire, de manière formelle, une fonction

$$S_{\text{mix}}(\mathcal{E}_{\text{rule}}, \mathcal{E}_{\text{data}}) = F(\text{logic}(\theta_{\text{rule}}), \text{embedding}(\mathbf{x}_{\text{data}})),$$

où **logic** exprime la partie symbolique (règles, connecteurs logiques, etc.), tandis que **embedding** encode la portion sub-symbolique. La fonction  $F$  peut évaluer une **co-information** ou un **score** d'adéquation entre la règle et les données. Cette démarche reprend les principes du **DSL** (sections 1.5.7.2 et 1.5.7.3) en donnant un **cadre** unifié pour la synergie entre une entité symbolique et une entité sub-symbolique.

Au fur et à mesure que le **nombre de règles** logiques augmente, on observe que la taille du **réseau** s'accroît, car chaque  $\mathcal{E}_{\text{rule}}$  ou  $\mathcal{E}_{\text{concept}}$  vient s'ajouter aux nœuds. Les **calculs** de synergie  $S_{\text{mix}}(\mathcal{E}_{\text{rule}}, \mathcal{E}_{\text{data}})$  peuvent devenir coûteux si l'on doit évaluer régulièrement de nombreuses règles sur de vastes flux sub-symboliques. Il est alors crucial de recourir à des **techniques de parsimonie**, en particulier celles décrites en section 1.4.5, qui permettent de **supprimer** ou de **couper** certaines liaisons  $\omega_{i,j}$  dès lors que leur pondération demeure faible. Cette régulation limite la prolifération de liens inutiles, maintenant la **complexité** à un niveau gérable.

Lorsque plusieurs **règles** se **contredisent** au sein du **SCN**, des **tensions** apparaissent. Il est possible que l'une des entités sub-symboliques (par exemple un flux de capteurs) présente une **synergie** positive avec une première règle, mais se heurte à une **incompatibilité** logique avec la seconde. Dans un tel cas, on peut introduire un **score de cohérence** ou une **pénalité**  $P(\mathcal{E}_{\text{rule}_1}, \mathcal{E}_{\text{rule}_2})$ , reflétant le fait que les règles  $\text{rule}_1$  et  $\text{rule}_2$  ne peuvent valablement être activées en même temps. On peut alors modifier l'évolution de chaque pondération afin de **réduire** la force de liens conduisant trop souvent à cette contradiction, ou introduire un **terme** correctif dans la mise à jour :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta [S_{\text{mix}}(\mathcal{E}_i, \mathcal{E}_j) - \tau \omega_{i,j}(t)] - \alpha P(\mathcal{E}_i, \mathcal{E}_j),$$

où  $\alpha$  contrôle l'intensité de la pénalité due à la contradiction.

D'un point de vue **implémentation**, les **limites** principales résident dans l'**explosion combinatoire** lorsque l'on souhaite intégrer non seulement des synergies binaires (section 1.4.4) mais également des synergies n-aires (section 1.4.7) et des **connaissances** logiques toujours plus nombreuses. De même, il peut être nécessaire de définir des **règles** plus modulaires ou paramétriques afin d'éviter la création d'une entité symbolique distincte pour chaque énoncé. Les entités pourraient alors coder des **schémas** logiques, associés à des **variables** substituables, pour diminuer le nombre total de nœuds.

Dans l'ensemble, ces **défis** mathématiques et de **mise en œuvre** ne remettent pas en cause le principe fondateur de l'approche hybride (sections 1.5.7.2 et 1.5.7.3), à savoir la possibilité de **fuser** la rigueur symbolique avec la force d'apprentissage sub-symbolique. Ils incitent plutôt à développer des **algorithmes** de parsimonie adaptés, à concevoir des mécanismes de **cohérence** entre règles, et à prévoir des **heuristiques** ou **méthodes** de clustering pour gérer la croissance du réseau. Le **DSL**, dans ce contexte, se distingue par sa **capacité** d'**auto-organisation** : il veille à ce que seules les **liaisons**  $\omega_{i,j}$  d'une **synergie** suffisante se stabilisent, et il répartit de façon **adaptive** les efforts de calcul selon les **bénéfices** perçus.

### 1.5.7.6. Conclusion

L'**incorporation** de composantes **symboliques** ou **cognitives** au sein du **Deep Synergy Learning (DSL)** ouvre un champ d'**IA hybride** dans lequel la **partie sub-symbolique** (traitement de données massives, bruitées et hétérogènes) est **complétée** par une **dimension symbolique** (règles, concepts, ontologies et raisonnement explicite). Cette combinaison se déploie naturellement dans le **Synergistic Connection Network (SCN)** qui, grâce à ses entités adaptatives et à ses liaisons synergiques évolutives, apparaît particulièrement adapté pour réunir et faire coévoluer ces deux approches.

On peut modéliser la **synergie** entre une **règle**  $R$  et un **module perceptif**  $\mathcal{E}_{\text{data}}$  à l'aide d'une fonction

$$S(\mathcal{E}_{\text{rule}}, \mathcal{E}_{\text{data}}) = f(\theta_{\text{rule}}, \mathbf{x}_{\text{data}}),$$

où  $\theta_{\text{rule}}$  encode l'information symbolique (logique, ontologie...) et  $\mathbf{x}_{\text{data}}$  représente les **features** sub-symboliques. La pondération  $\omega_{\text{rule}, \text{data}}(t)$  associée se met alors à jour de façon adaptative, selon

$$\omega_{\text{rule}, \text{data}}(t+1) = \omega_{\text{rule}, \text{data}}(t) + \eta [S(\mathcal{E}_{\text{rule}}, \mathcal{E}_{\text{data}}) - \tau \omega_{\text{rule}, \text{data}}(t)].$$

Cette dynamique assure que seules les **connexions** réellement bénéfiques (au sens de la synergie mesurée) se consolident, et permet à des **clusters** associant règles symboliques et entités sub-symboliques d'émerger spontanément au sein du réseau.

Ce mécanisme procure de **multiples avantages**. Il **préserve** la capacité d'**apprentissage** sub-symbolique, cruciale pour dompter des données riches et bruitées, tout en **introduisant** des notions symboliques aisément **explicitables** (règles, concepts, axiomes). Cette alliance favorise une **explication** plus claire des décisions, dans la mesure où la conclusion d'un cluster peut être reliée à la **règle**  $R$  (logique) et au **signal**  $\mathbf{x}$  (apprentissage neuronal). De plus, le DSL conserve ses forces intrinsèques, telles que la **multimodalité**, l'**adaptation** continue face aux variations de distribution et la **résilience** aux données partielles ou contradictoires.

Il faut toutefois noter qu'en **élargissant** la base de **règles** ou de **concepts**, le réseau peut s'agrandir et accroître la **complexité** de calcul, comme évoqué en section 1.5.7.5. Des mécanismes de **parsimonie** ou de **régulation** (limitation des connexions inutiles, seuils dynamiques sur les liaisons) demeurent essentiels pour éviter une explosion combinatoire. On peut également ajouter des **pénalités** ou un **score de cohérence** afin de gérer des contradictions logiques entre plusieurs règles.

En définitive, on obtient ainsi un **DSL** apte à **raisonner** en s'appuyant sur sa **base symbolique**, tout en **apprenant** de manière continue via sa **dimension sub-symbolique**. Cette architecture apparaît comme un **paradigme plus complet**, se rapprochant de la **cognition humaine**, laquelle jongle entre des **concepts** explicites et des **associations** perceptives implicites. Les sections et principes exposés dans ce chapitre (1.5) soulignent la **flexibilité** du DSL et son aptitude à jeter des **ponts** entre la **puissance** d'apprentissage neuronale et la **lisibilité** de la connaissance symbolique.



## I. 5 Exercices Simples

### Exercice 1 : Synergie vs. Architecture Figée

(Lien avec 1.5.1)

32. Rappelez ce qu'on entend par "architecture figée" dans un réseau neuronal profond (quel est son principe ?).
33. Expliquez en quoi une approche "synergique" (via des liens adaptatifs entre entités) peut rendre le réseau plus flexible sans modifier l'architecture d'ensemble.
34. Donnez un exemple concret où un réseau hiérarchique classique serait limité, alors qu'une approche DSL pourrait se reconfigurer spontanément.

### Exercice 2 : Multi-modalité Naturelle

(Lien avec 1.5.2)

35. Qu'est-ce qu'on entend par "fusion tardive" et "fusion précoce" dans les approches multi-modales traditionnelles ?
36. Comparez ces méthodes à la "co-intégration libre" dans le DSL : en quoi la synergie facilite-t-elle la découverte de liens entre modalités ?
37. Proposez un cas (vision + audio + texte) où la multi-modalité est cruciale, et indiquez comment le DSL l'absorbe sans configurer manuellement une "couche de fusion".

### Exercice 3 : Données Manquantes ou Bruitées

(Lien avec 1.5.3)

38. Dans un pipeline classique, citez au moins deux méthodes usuelles pour traiter les données incomplètes.
39. Expliquez la notion d'"auto-exclusion" d'une entité bruyante dans le DSL (comment se manifeste la chute de synergie ?).
40. Donnez un exemple imagé montrant comment un capteur défaillant peut être automatiquement mis à l'écart par l'auto-organisation synergique.

### Exercice 4 : Adaptation Continue et Lifelong Learning

(Lien avec 1.5.4)

41. Définissez le "catastrophic forgetting" dans un réseau neuronal classique.
42. Montrez en quelques phrases pourquoi le DSL, avec ses liens ajustés en permanence, peut mieux gérer l'arrivée d'une nouvelle tâche ou d'un nouveau flux de données.
43. Donnez un exemple où l'on ajoute une entité  $\mathcal{E}_{\text{nouvelle}}$  en milieu de parcours : comment se forment les connexions si la synergie est positive ?

### Exercice 5 : Symbolique et Sub-symbolique

(Lien avec 1.5.7)

44. Qu'entend-on par "entité symbolique" vs. "entité sub-symbolique" dans un réseau ?
45. Donnez un exemple de règle symbolique qui pourrait être encodée dans une entité  $\mathcal{E}_{\text{règle}}$ .
46. Décrivez brièvement comment cette règle peut se lier à une entité perceptive  $\mathcal{E}_{\text{vision}}$  si un motif détecté correspond à la condition de la règle.

## II. 5 Grands Problèmes Approfondis

### Problème 1 : Avantages Structuraux et Comparaison aux Réseaux Neuronaux

(Couvre 1.5.1, avec références à 1.5.3 et 1.5.4 éventuellement)

#### Énoncé

On veut analyser la différence entre une architecture figée de réseau profond et l'approche "auto-organisée" du DSL pour un scénario de classification complexe (images + quelques méta-informations).

#### 47. Question A :

Décrivez comment, dans un CNN classique, les couches sont conçues et pourquoi cela peut rendre l'apprentissage rigide quand on introduit de nouvelles classes d'images ou de nouvelles caractéristiques (méta-informations).

#### 48. Question B :

Montrez, en termes de synergie, comment un bloc  $\mathcal{E}_{\text{métainfo}}$  peut s'auto-connecter à un bloc  $\mathcal{E}_{\text{vision}}$  si l'on détecte une forte complémentarité. Quelles conséquences sur la structure du réseau ?

#### 49. Question C :

Comparez la souplesse d'évolution du DSL lorsqu'on ajoute des "features" ou lorsqu'une partie du CNN classique doit être reconçue. Quelle conclusion tirer pour des tâches en flux continu (continual learning) ?

### Problème 2 : Multi-Modalité et Synergie dans un Système Réel

(Couvre 1.5.2, avec liens vers 1.5.6 : représentations riches)

#### Énoncé :

On se place dans un système de reconnaissance d'événements (audio + vidéo + texte). Décrivez comment les entités correspondantes se forment et coévoluent.

#### 50. Question A :

Définissez les entités  $\mathcal{E}_{\text{vid}}$ ,  $\mathcal{E}_{\text{aud}}$ ,  $\mathcal{E}_{\text{txt}}$ . Proposez une forme pour leur représentation interne (ex. vecteurs embedding).

#### 51. Question B :

Comment la synergie "audio-texte" peut-elle se mesurer ? Donnez une idée (distance, co-information, etc.). Qu'est-ce que cela implique pour la création de liens  $\omega_{\text{aud,txt}}$  ?

#### 52. Question C :

Supposez qu'un 4<sup>e</sup> flux apparaisse (capteur de mouvement). Quelles étapes se produisent pour que  $\mathcal{E}_{\text{mouv}}$  s'insère ou non dans le cluster multimodal formé ?

#### 53. Question D :

Montrez en quoi cette intégration multi-modale **améliore** la "richesse" de la représentation globale, et commentez l'aspect d'**interprétabilité** (section 1.5.6) : comment visualiser et expliquer un cluster audio-vidéo-texte-mouvement ?

### Problème 3 : Gestion du Bruit, Adaptation Continue et Réduction de Supervision

(Couvre 1.5.3, 1.5.4, 1.5.5)

#### Énoncé

On s'intéresse à un contexte industriel où plusieurs capteurs surveillent une chaîne de production. Certains capteurs sont souvent défaillants (bruit élevé), et on n'a que peu de labels pour repérer les "pannes" effectives. On veut voir pourquoi le DSL s'en sort mieux qu'un réseau neuronal classique.

#### 1. Question A :

Décrivez comment l'**auto-exclusion** d'un capteur bruyant se produit dans le DSL : quel rôle joue la baisse de synergie  $\omega_{i,j}$  ? Expliquez la dynamique mathématique (rappel de la mise à jour  $\omega \leftarrow \omega + \eta[S - \tau\omega]$ ).

2. **Question B :**

Illustrez la notion de "peu de labels" : comment le DSL peut structurer les capteurs et détecter des clusters anormaux (pannes potentielles) **sans** un ensemble massivement labellisé ?

3. **Question C :**

Montrez qu'en cas de changement de distribution (p. ex. nouvelle configuration de la chaîne), la **mise à jour continue** favorise la réorganisation progressive, alors qu'un réseau classique exigerait un réapprentissage partiel ou total (risque de catastrophic forgetting).

4. **Question D :**

Concluez en expliquant pourquoi cette approche diminue la **dépendance** à un jeu de labels exhaustif (mieux qu'un CNN supervisé standard).

#### Problème 4 : Représentations Interprétables et Conception d'un DSL "lisible"

(Couvre 1.5.6 : représentations riches et interprétables, avec aspects multi-modal / clusters)

**Énoncé :**

On veut mettre en place un DSL pour une application de recommandation de contenus (médias, articles, etc.). L'**interprétabilité** est cruciale : on souhaite comprendre pourquoi le réseau propose tel contenu à tel utilisateur.

1. **Question A :**

Proposez un modèle d'entités  $\{\mathcal{E}_{\text{user}}\}$  et  $\{\mathcal{E}_{\text{content}}\}$ . Quelles informations contiennent-elles (profils, features, historique) ?

2. **Question B :**

Expliquez la formation spontanée de **clusters** : des groupes d'utilisateurs + contenus qui partagent une synergie. Comment lire ces clusters pour en extraire des explications (ex. "ce cluster aime la musique rock, ce contenu est du rock, etc.") ?

3. **Question C :**

Peut-on imaginer un macro-cluster qui regroupe plusieurs sous-ensembles (genres musicaux, types de public) ? Montrez comment la synergie de second ordre (ou plus) peut révéler des regroupements plus larges.

4. **Question D :**

Mettez en avant la différence avec un système de recommandation standard (collaborative filtering) : en quoi la "coopération adaptative" du DSL facilite-t-elle l'explicabilité ? Donnez un exemple narratif.

#### Problème 5 : Intégration de Symbolique et de Sub-symbolique dans le DSL

(Couvre 1.5.7 : dimensions symboliques ou cognitives)

**Énoncé :** On cherche à intégrer un **module de règles logiques** dans un DSL qui, par ailleurs, gère des flux sub-symboliques (vision, audio). L'objectif : détecter un événement "danger" si plusieurs conditions symboliques et perceptives sont réunies.

1. **Question A :**

Donnez un **exemple** concret d'une règle symbolique  $\mathcal{E}_{\text{rule}}$  : "Si la température > 100°C ET la pression > 5 bars, ALORS suspecter surchauffe."

2. **Question B :**

Expliquez comment cette règle peut être une entité  $\mathcal{E}_{\text{rule}}$  dans le DSL, et quelle forme prend la synergie  $\omega_{\text{rule}, \text{capteurTemp}, \text{capteurPress}}$ .

3. **Question C :**

Décrivez la **formation** d'un cluster  $\{\mathcal{E}_{\text{rule}}, \mathcal{E}_{\text{capteurTemp}}, \mathcal{E}_{\text{capteurPress}}\}$ . Qu'advient-il si un 3<sup>e</sup> paramètre (ex. "vibration") s'ajoute ? Sous quelles conditions la règle évolue ou se connecte à cette nouvelle entité ?

4. **Question D :**

Quels **défis** (scalabilité, cohérence logique, etc.) se posent si on introduit plusieurs dizaines de règles symboliques dans le même DSL ? Comment gérer la contradiction entre deux règles ?

## Conclusion

Ces **5 exercices** et **5 grands problèmes** couvrent l'essentiel des **motifs** pour lesquels une approche **synergique** (DSL) se révèle attrayante :

- **Comparaison** avec les réseaux neuronaux profonds (1.5.1),
- **Gestion** fluide de la **multi-modalité** (1.5.2),
- **Souplesse** face aux données bruitées ou incomplètes (1.5.3),
- **Adaptation** continue ou auto-évolution (1.5.4),
- **Réduction** de la supervision (1.5.5),
- **Représentations** plus riches et interprétables (1.5.6),
- **Dimensions** symboliques ou cognitives (1.5.7).

Les exercices offrent une **entrée rapide**, tandis que les problèmes approfondissent chacun plusieurs aspects, mettant en avant la **dynamique** du DSL et sa **philosophie** de "co-apprentissage" par la **synergie**.

## I. 5 Exercices (courts, purement mathématiques)

### Exercice 1 : Mesure de Synergie et Point Fixe

(Référence à 1.5.1 : Avantages face aux réseaux profonds, via une pondération adaptative)

Soit une entité  $\mathcal{E}_i$  et une entité  $\mathcal{E}_j$ , dont la pondération synergique suit la règle :

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

avec  $\eta, \tau > 0$  et  $S_{i,j} \in \mathbb{R}$  une constante (mesure de synergie binaire).

1. **Montrez** qu'il existe un point fixe  $\omega_{i,j}^*$  pour cette dynamique et **calculez**  $\omega_{i,j}^*$ .
2. **Discutez** la condition de stabilité (linéarisation autour de  $\omega_{i,j}^*$ ) menant à  $\eta \tau < 1$ .
3. **Concluez** brièvement pourquoi cet ajustement local favorise la flexibilité structurelle, en comparaison d'un poids figé dans un réseau profond.

### Exercice 2 : Synergie Multi-modale par Distance

(Référence à 1.5.2 : multi-modalité)

On définit deux entités  $\mathcal{E}_A$  (domaine “visuel”) et  $\mathcal{E}_B$  (domaine “audio”), chacune représentée par un vecteur dans  $\mathbb{R}^d : \mathbf{x}_A, \mathbf{x}_B$ . On pose :

$$d(\mathbf{x}_A, \mathbf{x}_B) = \|\mathbf{x}_A - \mathbf{x}_B\|, \quad S(\mathcal{E}_A, \mathcal{E}_B) = \frac{1}{1 + \|\mathbf{x}_A - \mathbf{x}_B\|^2}.$$

1. **Prouvez** que  $S(\mathcal{E}_A, \mathcal{E}_B)$  est inversement proportionnelle à la distance euclidienne au carré.
2. **Montrez** que  $S(\cdot, \cdot)$  atteint son maximum quand  $\mathbf{x}_A = \mathbf{x}_B$ .
3. **Interprétez** mathématiquement la situation où  $\mathbf{x}_A$  et  $\mathbf{x}_B$  proviennent de modalités différentes : pourquoi  $S(\cdot, \cdot)$  peut rester élevé même si  $\mathbf{x}_A \neq \mathbf{x}_B$  au sens brut, mais “proches” dans un espace latent ?

### Exercice 3 : Complétion Partielle via Synergie

(Référence à 1.5.3 : flexibilité face aux données incomplètes)

Soient deux vecteurs  $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$ , chacun avec des composantes manquantes (codées par NaN ou \*). On définit un opérateur  $\oplus$  qui ignore les composantes manquantes et évalue la distance sur les seules dimensions communes. Par exemple :

$$d_{\oplus}(\mathbf{u}, \mathbf{v}) = \sqrt{\sum_{k \in I} (u_k - v_k)^2},$$

où  $I \subset \{1, \dots, d\}$  est l'ensemble des composantes non manquantes à la fois chez  $\mathbf{u}$  et  $\mathbf{v}$ .

1. **Formulez** une mesure de synergie  $S(\mathbf{u}, \mathbf{v})$  à partir de  $d_{\oplus}$ .
2. **Démontrez** que si  $\mathbf{u}$  et  $\mathbf{v}$  partagent un nombre restreint de dimensions, la variance sur ces dimensions peut suffire à créer un lien fort si la similarité est forte dessus.
3. **Commentez** la robustesse face au bruit : si  $\mathbf{u}$  est bruyante sur quelques composantes, pourquoi la définition par  $\oplus$  peut limiter l'influence de ces composantes défaillantes ?

### Exercice 4 : Dynamique Continue et Auto-Évolution

(Référence à 1.5.4 : adaptation continue)

On considère la version “continue” de l'équation de mise à jour :

$$\frac{d \omega_{i,j}}{dt} = \eta [S_{i,j} - \tau \omega_{i,j}(t)].$$

1. **Résolvez** l'EDO pour  $\omega_{i,j}(t)$  en supposant  $\omega_{i,j}(0) = \omega_0$ . Trouvez la forme explicite de la solution  $\omega_{i,j}(t)$ .
2. **Déterminez** la valeur limite lorsque  $t \rightarrow \infty$ .
3. **Interprétez** l'intérêt de cette dynamique dans un cadre “lifelong learning”, où  $\omega_{i,j}(t)$  peut s'ajuster en temps réel sans repasser par un entraînement batch global.

### Exercice 5 : Synergie Symbolique et Sub-symbolique

(Référence à 1.5.7 : dimensions symboliques)

On suppose deux types d'entités :

- $\mathcal{E}_{\text{rule}}$  (règle symbolique) décrite par un petit vecteur  $\mathbf{r} \in \{0,1\}^m$  (chaque composante représentant la présence d'une condition),
- $\mathcal{E}_{\text{feat}}$  (feature sub-symbolique) décrite par un vecteur réel  $\mathbf{x} \in \mathbb{R}^m$ .

Définissez une fonction de synergie :

1. **Proposez** une mesure  $S(\mathcal{E}_{rule}, \mathcal{E}_{feat})$  qui compare  $\mathbf{r}$  et  $\mathbf{x}$  (par ex. un produit mixte).
2. **Montrez** qu'on peut extraire une valeur de "compatibilité" logique, si  $\mathbf{r}_k = 1$  implique  $\mathbf{x}_k > \theta$ .
3. **Concluez** sur l'idée d'**hyper-arêtes** si l'on intègre plusieurs règles ou features simultanément.

## II. 5 Grands Problèmes (avec plusieurs questions mathématiques)

### Problème 1 : Analyse Formelle des Avantages face aux Réseaux Profonds

(Couvre 1.5.1, éléments de 1.5.3 et 1.5.4)

#### Énoncé mathématique :

On modélise un réseau neuronal "classique" (à architecture figée) comme un ensemble de matrices de poids  $\{W^{(l)}\}_{l=1}^L$ . Pour incorporer un nouveau module ou une nouvelle tâche, il faut ajouter des poids  $\Delta W$  et réentraîner partiellement. Dans le **DSL**, on a une matrice  $\Omega(t)$  de taille  $n \times n$ , évoluant selon :

$$\Omega(t+1) = \Omega(t) + \eta[S(\Omega(t)) - \tau \Omega(t)],$$

pour une fonction "globale"  $S(\cdot)$ .

1. **Question A** (Comparaison paramétrique) :

Évaluez la **dimension paramétrique** d'un CNN figé (où seule la rétropropagation ajuste  $\{W^{(l)}\}$ ) contre un DSL pouvant **augmenter ou réduire** la taille de  $\Omega(t)$  (nouvelles entités, etc.). Quelle conclusion sur la capacité d'évolution ?

2. **Question B** (Stabilité d'un point fixe global) :

Supposons qu'on définisse une fonction énergie  $\mathcal{J}(\Omega) = -\sum_{i,j} \Omega_{i,j} S_{i,j} + \text{termes de régularisation}$ . Montrez que la descente locale en  $\mathcal{J}$  entraîne la **création** ou la **dissolution** de liens. Comment cela surpasse la rigidité d'un CNN ?

3. **Question C** (Scénario d'ajout d'entités) :

Mathématisez la situation où on ajoute  $\Delta n$  entités, agrandissant  $\Omega$ . Définissez les conditions sous lesquelles ces nouvelles entités se connectent rapidement à des clusters existants.

### Problème 2 : Multi-modalité et Mesure de Synergie Non Linéaire

(Couvre 1.5.2, aspects 1.5.6)

#### Énoncé mathématique :

Soit un jeu de deux modalités  $\mathbf{x}_{vis} \in \mathbb{R}^d$  (image) et  $\mathbf{x}_{aud} \in \mathbb{R}^p$  (spectre audio). On veut définir une **co-information** n-aire :

$$I(\mathbf{x}_{vis}, \mathbf{x}_{aud}) = H(\mathbf{x}_{vis}) + H(\mathbf{x}_{aud}) - H(\mathbf{x}_{vis}, \mathbf{x}_{aud}),$$

puis en déduire une synergie.

1. **Question A** (Discrétisation) :

Montrez que si on approxime  $\mathbf{x}_{vis}$  et  $\mathbf{x}_{aud}$  par des variables discrètes (binning, par ex.), la **complexité** de l'estimation d'entropie croît exponentiellement avec la dimension.

2. **Question B** (Définition de la synergie) :

Proposez une fonction  $S(\mathbf{x}_{vis}, \mathbf{x}_{aud}) = \max(0, I(\mathbf{x}_{vis}, \mathbf{x}_{aud}))$ . Montrez que cette synergie devient plus grande si les deux modalités sont statistiquement dépendantes.

3. **Question C** (Clusterisation auto-organisée) :

Supposez que  $\mathbf{x}_{vis}, \mathbf{x}_{aud}, \mathbf{x}_{txt}$  forment un trio. Comment formuler la synergie n-aire

$$S_3(\mathbf{x}_{vis}, \mathbf{x}_{aud}, \mathbf{x}_{txt})$$

en termes d'information mutuelle conjointe ? Quelles difficultés (estimation, calcul) souligner ?

### Problème 3 : Données Incomplètes, Bruit et Équations de Mise à Jour

(Couvre 1.5.3, 1.5.4, 1.5.5)

#### Énoncé mathématique :

Soit un ensemble de vecteurs  $\{\mathbf{x}_i\}_{i=1}^n$  dans  $\mathbb{R}^d$ , dont certains composants sont manquants ou bruités. On définit une **matrice de similarités**  $[S_{i,j}]$  adaptée : si  $\mathbf{x}_i$  ou  $\mathbf{x}_j$  sont incomplètes, on calcule la similarité sur les dimensions communes. On met à jour  $\omega_{i,j}(t)$  par

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

1. **Question A** (Blocage par bruit) :

Définissez un seuil  $\omega_{\min}$ . Montrez que si  $\mathbf{x}_k$  est très bruyante, son  $\omega_{k,j}(t)$  pour tout  $j$  reste durablement  $< \omega_{\min}$ . En déduire l'**isolement** de  $\mathcal{E}_k$ .

2. **Question B** (Renforcement partiel) :

Montrez qu'une entité  $\mathcal{E}_m$  partiellement incomplète peut quand même obtenir un  $\omega_{m,j}$  élevé si sur les dimensions disponibles (communes), la similarité est grande.

3. **Question C** (Inclusion dans un cluster) :

Formalisez la condition d'entrée de  $\mathcal{E}_m$  dans un cluster  $\mathcal{C} \subset \{1, \dots, n\}$  : la somme interne  $\sum_{j \in \mathcal{C}} \omega_{m,j}$  doit dépasser un seuil. Discutez de l'aspect "faible supervision".

### Problème 4 : Représentations Riches, Clusters et Macro-Clusters

(Couvre 1.5.6, aspects 1.5.2)

#### Énoncé mathématique :

On suppose qu'on a un ensemble d'entités  $\{\mathcal{E}_i\}_{i=1}^n$ , chacune pouvant posséder sa représentation  $\mathbf{x}_i \in \mathbb{R}^d$ . La synergie binaire est

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

pour une fonction  $f$  quelconque (distance, similarité, co-information). On définit un **cluster**  $\mathcal{C}$  comme un sous-ensemble maximisant

$$\Sigma(\mathcal{C}) = \sum_{i,j \in \mathcal{C}} \omega_{i,j}.$$

1. **Question A** (Macro-cluster) :

Introduisez une fonction  $\Omega(\mathcal{C}_1, \mathcal{C}_2) = \frac{1}{|\mathcal{C}_1||\mathcal{C}_2|} \sum_{i \in \mathcal{C}_1, j \in \mathcal{C}_2} \omega_{i,j}$ . Montrez en termes algébriques comment la "fusion"  $\mathcal{C}_1 \cup \mathcal{C}_2$  se produit si  $\Omega(\mathcal{C}_1, \mathcal{C}_2)$  dépasse un certain seuil.

2. **Question B** (Explicabilité via sous-graphes) :

Supposez qu'on regarde la matrice  $\Omega$  restreinte à un cluster  $\mathcal{C}$ . Commentez la diagonale bloquée  $\{\omega_{i,i}\}_{i \in \mathcal{C}}$ . Montrez qu'une structure en "sous-blocs" peut indiquer des **macro-clusters** successifs.

3. **Question C** (Richesse des représentations) :

Expliquez pourquoi des entités diverses (ex. vecteurs image, vecteurs audio, entités symboliques) induisent des clusters hétérogènes, et discutez l'avantage d'avoir plusieurs macro-clusters plutôt qu'un unique super-cluster.

## Problème 5 : Entités Symboliques, Synergie Logique et Sub-symbolique

(Couvre 1.5.7)

### Énoncé mathématique :

On introduit un ensemble d'entités logiques  $\{\mathcal{E}_\alpha^{(\text{symb})}\}_{\alpha=1}^r$  et un ensemble d'entités sub-symboliques  $\{\mathcal{E}_\beta^{(\text{feat})}\}_{\beta=1}^s$ . Pour toute règle symbolique  $\mathcal{R}_\alpha$ , on l'encode comme un vecteur binaire  $\mathbf{r}_\alpha \in \{0,1\}^d$ . Les features  $\mathbf{x}_\beta \in \mathbb{R}^d$ . On définit :

$$S(\mathcal{R}_\alpha, \mathcal{E}_\beta^{(\text{feat})}) = \sum_{k=1}^d [\mathbf{r}_\alpha[k] \cdot \phi(\mathbf{x}_\beta[k])],$$

Où  $\phi(\cdot)$  est une fonction d'activation assurant un gain si la condition binaire est satisfaite.

1. **Question A** (Formulation du produit logique) :

Montrez qu'en prenant  $\phi(u) = \mathbf{1}_{\{u > \theta\}}$  (indicateur), la synergie est un compte du nombre de dimensions  $k$  où  $\mathbf{r}_\alpha[k] = 1$  ET  $\mathbf{x}_\beta[k] > \theta$ .

2. **Question B** (Évolution des liaisons) :

Écrivez l'équation

$$\omega_{\alpha,\beta}(t+1) = \omega_{\alpha,\beta}(t) + \eta[S(\mathcal{R}_\alpha, \mathcal{E}_\beta^{(\text{feat})}) - \tau \omega_{\alpha,\beta}(t)].$$

Montrez que si la règle symbolique  $\mathcal{R}_\alpha$  s'applique souvent aux features  $\mathbf{x}_\beta$ , la pondération  $\omega_{\alpha,\beta}$  se consolide.

3. **Question C** (Clusters mixtes) :

Définissez la notion d'un **cluster** contenant à la fois des règles logiques et des entités de features. Montrez comment la somme interne  $\sum_{\alpha,\beta \in \mathcal{C}} \omega_{\alpha,\beta}$  peut révéler un module cognitivement lisible (logique + perception).

4. **Question D** (Limite d'échelle) :

Discutez la difficulté si le nombre total de règles  $\mathbf{r}_\alpha$  et d'entités  $\mathbf{x}_\beta$  est très grand. Quels compromis doit-on envisager pour réduire la complexité ?



## 1.6. Applications Pressenties et Domaines Impactés

Le **Deep Synergy Learning (DSL)**, en tant que paradigme alliant **auto-organisation**, **synergies adaptatives** et **capacité d'évolution continue**, suscite un **intérêt grandissant** dans de nombreux champs de l'intelligence artificielle. Après avoir dressé, dans les sections précédentes, les fondements conceptuels et structurels du DSL (sections 1.1 à 1.5), il est temps de se projeter vers les **applications** et les **domaines** où cette approche pourrait exercer un **impact fort**.

L'architecture **distribuée** et **plastique** du DSL, la possibilité d'**auto-organiser** des entités multimodales ou symboliques, ainsi que sa **tolérance** aux données bruitées, confèrent au DSL un potentiel d'**adaptation** et de **complémentarité** qui dépasse souvent les limites des modèles traditionnels (réseaux neuronaux profonds strictement hiérarchiques ou méthodes symboliques pures). De la **vision artificielle** à la **robotique**, en passant par la **reconnaissance audio**, la **recommandation personnalisée**, la **surveillance** ou encore la **planification industrielle**, le DSL offre des réponses nouvelles, mieux adaptées à la complexité des flux de données et aux exigences d'évolution en temps réel.

Dans cette section (1.6), nous présentons un **éventail** des applications pressenties, non seulement comme une liste d'exemples, mais aussi comme une **cartographie** des domaines que le DSL peut transformer :

5. **Vision Artificielle et Reconnaissance d'Objets Complexes** (1.6.1)
6. **Analyse Audio et Traitement du Langage Naturel** (1.6.2)
7. **Robotique et Systèmes Intelligents Évolutifs** (1.6.3)
8. **Recommandation Personnalisée et Systèmes de Décision** (1.6.4)
9. **Surveillance, Diagnostic Médical et Anomalies** (1.6.5)
10. **Planification et Optimisation dans l'Industrie 4.0** (1.6.6)
11. **Perspectives pour la Recherche Fondamentale en IA Forte** (1.6.7)

Pour chacun de ces thèmes, le DSL se révèle prometteur, en raison de ses **mécanismes** d'auto-organisation : **clusters dynamiques**, **pondérations adaptatives**, **intégration** de multiples sources (visuelles, auditives, textuelles, capteurs...), et **apprentissage** continu sans dépendance exclusive à la supervision. Nous verrons ainsi comment le DSL peut contribuer à relever certains **défis** actuels de l'IA, tels que la **plasticité**, la **résilience** aux données incertaines, ou encore la **capacité** à modéliser la **complexité** grandissante des problèmes.

### 1.6.1. Vision Artificielle et Reconnaissance d'Objets Complexes

Parmi les champs qui ont le plus bénéficié de l'essor des réseaux neuronaux profonds, on trouve la **vision artificielle**. Les **CNN** (Convolutional Neural Networks) ont acquis une renommée considérable pour la détection et la classification d'objets dans des images. Toutefois, les tâches de **reconnaissance** deviennent de plus en plus **subtiles** : il ne s'agit plus seulement d'identifier un objet (ex. un chat, un chien), mais de **détecter** et de **comprendre** des **scènes complexes**, dans des environnements qui évoluent, avec des conditions lumineuses variables, des objets partiellement masqués ou des articulations multiples d'objets. Le **Deep Synergy Learning** ouvre ici de nouvelles perspectives, dont nous détaillons ci-après les ressorts principaux.

#### 1.6.1.1. Au-delà d'une Hiérarchie Rigide : Clusters d'Entités Visuelles

Dans un **CNN** classique, les couches de convolution et de pooling s'enchaînent selon un **pipeline** prédéfini : la première couche repère des motifs élémentaires (bords, contrastes), la seconde assemble ces bords pour former des traits plus élaborés, et ainsi de suite, jusqu'à la classification. Bien que très performant, ce mécanisme demeure relativement **rigide**. Il impose un ordre précis selon lequel les "features" visuelles doivent être détectées, sans autoriser la création ou la suppression dynamique de filtres ni de liaisons entre diverses régions de l'image.

Le **Deep Synergy Learning (DSL)** envisage les **features** ou régions d'images comme autant d'**entités**  $\mathcal{E}_i$ , chacune portant un **descripteur** lié à un patch, un certain motif ou un vecteur plus abstrait (par exemple issu d'un autoencodeur local). Ces entités visuelles s'**auto-organisent** en **clusters** dès qu'elles identifient une **synergie** notable (proximité de descripteurs, co-occurrences de motifs, etc.). L'**hiérarchie** (ou la "stratification") n'est plus imposée par des couches fixes, mais émerge au contraire de la **coopération** entre entités qui jugent profitable leur rapprochement. On peut par exemple voir se constituer un cluster dédié aux formes circulaires, un autre regroupant les zones de fortes lignes horizontales, et d'autres encore, voire des **macro-clusters** combinant plusieurs de ces groupes si leur fusion devient pertinente.

Cette **dynamique** apporte une plus grande **flexibilité** dans la reconnaissance d'objets complexes ou inattendus. Si un objet n'appartient pas aux catégories standards (par exemple, un objet partiellement masqué ou inconnu), les entités  $\mathcal{E}_k$  décrivant ses parties peuvent spontanément **coopérer** avec celles décrivant d'autres objets partiellement similaires. Il se crée alors un **sous-réseau** (cluster) susceptible de fusionner avec un groupement plus vaste, agrandissant le champ d'interprétation.

Pour un modèle plus formel, on peut représenter chaque entité visuelle  $\mathcal{E}_k$  par un vecteur  $\mathbf{x}_k \in \mathbb{R}^d$ . La synergie, qu'elle soit définie via une **distance** (ex. inversée) ou une **co-information**, détermine la valeur  $S(\mathcal{E}_k, \mathcal{E}_m)$ . Les **pondérations**  $\omega_{k,m}$  s'ajustent de façon adaptative au fil des données. Plusieurs entités corrélées finissent par **stabiliser** leurs liaisons internes, formant un **cluster** autour d'un même objet, d'un motif, ou d'une scène particulière. Cette organisation n'est pas définitive : si le réseau découvre que d'autres entités apportent une information complémentaire, les **liaisons** s'amplifient, élargissant ou refaçonnant le **sous-graphe** pour intégrer de nouvelles découvertes visuelles. En ce sens, la reconnaissance s'établit de manière plus **organique** qu'avec un pipeline hiérarchique statique, autorisant une diversité de regroupements et le partage de sous-patterns entre plusieurs objets si c'est jugé profitable.

#### 1.6.1.2. Multi-Échelle et Robustesse aux Déformations

Dans le traitement d'objets complexes, il est fréquent de rencontrer des **variations** d'échelle (petit/grand dans l'image), des **transformations** géométriques (rotation, vue partielle, ombre) ou une **composition** de sous-parties distinctes. Les réseaux de neurones de type **CNN** gèrent généralement ces phénomènes en pratiquant la **convolution multi-échelle** ou la **data augmentation**, où l'on applique diverses transformations (zoom, rotation...) lors de l'entraînement. Cette stratégie, efficace dans bien des cas, reste toutefois largement paramétrée à l'avance (couches de pooling fixes, patterns choisis pour l'augmentation).

Le **Deep Synergy Learning (DSL)** propose un mécanisme différent. Des **entités**  $\mathcal{E}_i$  peuvent décrire un même objet à des échelles ou des angles divers, et leurs **liaisons**  $\omega_{i,j}$  se renforcent si la **synergie** (similarité, co-information) confirme qu'il s'agit bien de la **même structure** vue sous différentes perspectives. L'existence d'une entité dédiée à la "version miniaturisée" de l'objet et une autre à la "version grand format" aboutit à un **cluster** commun si elles trouvent un **gain** à leur rapprochement, sans exiger qu'on impose explicitement un niveau de pooling ou un tableau de transformations prédéterminé.

Un premier **exemple** (lié à l'invariance d'échelle) peut être formalisé en introduisant, pour un patch visuel  $\mathbf{x}_k$ , une entité  $\mathcal{E}_{k,\alpha}$  décrivant ce patch à l'échelle  $\alpha$ . Les pondérations  $\omega_{(k,\alpha),(k,\beta)}$  s'élèvent si la ressemblance (ou co-information) entre la représentation à l'échelle  $\alpha$  et à l'échelle  $\beta$  demeure élevée, ce qui crée un **cluster** multi-échelle au sein duquel ces entités se regroupent.

Un second **exemple** (visant les déformations) consiste à considérer  $\mathcal{E}_{k,\theta}$  comme la **version** du patch  $\mathbf{x}_k$  après avoir subi une **rotation**  $\theta$ . Deux angles  $\theta_1$  et  $\theta_2$  peuvent alors mener à des entités jugées "similaires", renforçant la liaison  $\omega_{(k,\theta_1),(k,\theta_2)}$ . On peut ainsi constituer un **macro-cluster** stable, reliant divers points de vue d'un même objet, garantissant une forme d'**invariance** à la rotation.

Cette approche confère une **robustesse** aux changements d'échelle et de pose. Le réseau n'a pas besoin de couches de pooling ni de transformations explicites ajoutées artificiellement : le **DSL** laisse les entités correspondantes **se découvrir** et **s'associer** si elles identifient un gain de synergie. L'ajustement se fait localement et **auto-organise** la reconnaissance, en évitant un pipeline rigide, pour composer des structures plus **adaptatives** et résilientes.

#### 1.6.1.3. Interaction avec d'Autres Modalités (Vision + ...)

Dans un cadre multimodal (voir section 1.5.2), la **vision** artificielle s'avère fréquemment complétée par un **flux audio** ou **textuel**. Le **Deep Synergy Learning (DSL)** n'établit pas de cloison pour la coopération entre l'entité visuelle  $\mathcal{E}_{\text{vis}}$  et l'entité auditive  $\mathcal{E}_{\text{aud}}$ . Le fait qu'elles représentent deux modalités différentes ne constitue pas une barrière : si la **synergie**  $S(\mathcal{E}_{\text{vis}}, \mathcal{E}_{\text{aud}})$  se révèle positive (corrélation ou co-information notable), les **pondérations**  $\omega_{\text{vis}, \text{aud}}$  se consolident, jusqu'à former un cluster dédié.

Dans la **reconnaissance** d'objets complexes, un **événement** visuel (comme un objet qui tombe) peut se coupler à un **événement** sonore simultané (bruit d'impact). Si la représentation  $\mathbf{x}_{\text{vis}}$  associée à la chute et la représentation  $\mathbf{x}_{\text{aud}}$  de l'onde sonore se rapprochent au sens de la **coopération**, un **sous-réseau**  $\{\mathcal{E}_{\text{patch}}, \mathcal{E}_{\text{featureAudio}}\}$  se constitue. Il se spécialise dans la détection de la scène "objet tombant plus choc acoustique", ce qui accroît les facultés de **classification** ou de compréhension de la situation ("Cassure", "Chute de vaisselle", etc.).

Cette **flexibilité** se généralise à d'autres flux, comme l'analyse **textuelle** (par exemple des légendes associées, des mots clés émis par un utilisateur), aboutissant à des micro-clusters multi-entités : la vision se combine à l'audio, et tous deux interagissent avec des entités textuelles. La dynamique d'**auto-organisation** du **DSL** détermine les **groupements** les plus profitables, sans exiger qu'un concepteur définisse un module de fusion dédié pour le couple "vision + audio". Cela favorise une **intégration** sans couture des diverses modalités, conduisant à une représentation plus riche et plus **résiliente** aux variations d'une modalité prise isolément.

#### 1.6.1.4. Impact sur les Problèmes de Vision Avancés

Dans le cadre de la **vision par ordinateur**, un **Deep Synergy Learning (DSL)** structuré en entités visuelles présente un atout certain pour traiter des scénarios avancés qui outrepassent la simple classification d'images. L'**auto-organisation** autorise l'émergence de **clusters** plus flexibles, allant au-delà du pipeline statique imposé par les méthodes classiques.

Dans la **détection d'objets multiples et partiellement superposés**, par exemple, un réseau traditionnel prévoit souvent un ensemble rigide d'ancrages (bounding boxes) ou un module de segmentation spécifique. Au contraire, le **DSL** peut laisser chaque patch ou région **s'affilier** à des **entités** "chien" et "chaise" même si ces objets se chevauchent dans l'image. Les pondérations  $\omega_{i,j}$  augmentent dès que

la synergie entre patchs associant l'idée de "chien" et l'idée de "chaise" se justifie. Le réseau n'a pas besoin d'un pipeline imposant a priori la détection séparée de chaque catégorie ; la constitution du **cluster** facilite l'identification conjointe de plusieurs objets.

Dans la **segmentation contextuelle**, on peut envisager que certaines entités segmentent l'image en superpixels et d'autres identifient des **contours**. Si leur **synergie** révèle une cohérence — par exemple, les bordures détectées coïncident avec les frontières d'un superpixel —, la liaison  $\omega$  se renforce. On forme alors un **cluster** correspondant à une "zone cohérente" dans l'image, laquelle pourrait se rattacher à un objet complet ou à une portion d'arrière-plan homogène. Cette approche s'écarte de la segmentation par apprentissage end-to-end, en permettant une coopération plus libre entre entités spécialisées (détection de bordures, regroupement de pixels...) selon la présence d'un **gain**.

Dans un contexte de **vision évolutive en robotique**, un agent dans un environnement changeant (luminosité, décor, nouveaux éléments) doit s'ajuster en continu. Un réseau de neurones traditionnel exigerait soit un réentraînement, soit une réinitialisation partielle. Le **DSL**, quant à lui, se contente de réévaluer localement la **synergie** entre entités visuelles : les connexions correspondant à d'anciennes conditions (ex. forte luminosité) se voient **décroître** si elles ne s'avèrent plus utiles, et de nouvelles entités ou liaisons peuvent apparaître pour gérer des éclairages inédits, des angles de vue différents ou de nouveaux objets. Cette **adaptation** locale maintient la **structure** globale du réseau, sans nécessiter un apprentissage complet de bout en bout, et confère une plasticité plus grande à la **vision** robotique, tout en conservant la possibilité d'exploiter ses acquis antérieurs.

#### 1.6.1.5. Conclusion Partielle : Une Alternative Complémentaire au CNN

Les **CNN** constituent toujours un **vecteur** extrêmement puissant pour la **vision** artificielle, en particulier lorsqu'un large corpus d'exemples annotés est disponible et que l'on vise des tâches de classification ou de détection sur des données stables. Toutefois, la structure même d'un **CNN** s'avère déterministe : chaque couche est fixée, et l'apprentissage se focalise principalement sur une logique de reconnaissance supervisée. Le **Deep Synergy Learning (DSL)** apporte une approche plus **organique** et **dynamique**, où les **entités** associées à des *patches* ou des *features* visuelles peuvent librement se **synergiser**, se **séparer**, s'**associer** avec d'autres modalités ou s'adapter aux **conditions** changeantes (nouvelles échelles, bruit, objets inédits).

La **structure** du **Synergistic Connection Network** autorise ainsi une **auto-organisation** continue, évitant la rigidité d'un pipeline imposé pour la vision. On peut introduire de nouvelles entités  $\mathcal{E}_{\text{nouvelle}}$  si un objet inconnu apparaît ou si l'on veut traiter un autre type de patch visuel, ce qui met à jour localement les connexions  $\omega$ . Rien n'empêche de **s'hybrider** avec un CNN : les **descripteurs** issus d'une couche intermédiaire d'un CNN peuvent constituer des entités  $\mathcal{E}_{\text{CNN}}$  dans le DSL, et la **synergie** entre ces entités reflète leur complémentarité.

Pour la **reconnaissance d'objets complexes** dans des scènes variées (éclairage fluctuant, angles de vue inédits, transformations géométriques), le **DSL** fournit un **cadre** mathématique structuré, avec des **équations** décrivant l'évolution des liaisons  $\omega_{i,j}$  et des mesures de synergie multi-échelle (ou non linéaire). Il enrichit la **vision** d'une dose de **plasticité** et de **robustesse**, sans empêcher l'usage de CNN comme base de features. Les **clusters** qui émergent dans ce réseau renvoient à des regroupements de patches ou de motifs liés par leur **gain mutuel**, offrant un point de vue plus distribué et collaboratif qu'un pipeline hiérarchique unique. Au final, le **DSL** s'érige en complément précieux aux **CNN** pour atteindre une **vision** flexible, résiliente et progressivement évolutive.

#### 1.6.2. Analyse Audio et Traitement du Langage Naturel

Après avoir vu comment le **Deep Synergy Learning (DSL)** peut apporter une flexibilité et une auto-organisation en **vision artificielle** (1.6.1), il est naturel de s'intéresser aux **données audio** et au **langage naturel**, deux domaines où la richesse des signaux et la variabilité contextuelle sont particulièrement élevées. Dans un réseau neuronal profond classique, les approches d'**analyse audio** (speech recognition, classification de sons, détection d'événements acoustiques) et de **traitement du langage naturel** (NLP) (analyse sémantique, traduction, question-réponse) demeurent puissantes, mais reposent souvent sur des modèles séparés (RNN, LSTM, Transformers) et des schémas d'apprentissage supervisé imposant de gros volumes de données annotées.

Le **DSL**, en revanche, propose une **auto-organisation** où chaque entité  $\mathcal{E}_i$  (issue de features audio ou textuelles) peut s'insérer dans un **réseau synergique** plus large, permettant :

12. Une **fusion plus organique** entre les éléments acoustiques ou linguistiques,
13. Une **adaptation continue** au bruit, aux accents, aux registres de langage,
14. Une cohabitation **symbolique–subsymbolique** pour des tâches de logique linguistique plus avancées.

Cette sous-section (1.6.2) détaille en quoi le DSL modifie l'approche habituelle de l'analyse audio et du traitement du langage, et en quoi il répond à des défis d'**hétérogénéité**, de **bruit**, de **variabilité linguistique** et de **scénarios évolutifs**.

### 1.6.2.1. Analyse Audio : Entités Sonores et Synergie Adaptative

Dans un **réseau** neuronal **classique** pour l'audio (CNN ou RNN appliqué aux spectrogrammes), on spécifie dès le départ un schéma d'extraction de *features* (MFCC, log-mel, etc.), puis l'on optimise un classifieur (par exemple, pour la reconnaissance de phonèmes ou la classification de sons). Le **Deep Synergy Learning (DSL)** introduit une logique plus **organique** en laissant des entités  $\mathcal{E}_{\text{aud},k}$  se constituer pour chaque **fenêtre** ou **segment** temporel (frames audio), pour un **spectre** fréquentiel spécifique, ou pour un **embedding** plus abstrait (mots, bruits, etc.). Ces entités se regroupent en **clusters** dès lors qu'elles observent un **gain** à coopérer, qu'il s'agisse de la ressemblance spectrale, de la proximité temporelle ou de la récurrence d'un même motif acoustique.

Si on modélise deux segments audios  $\mathbf{a}_i, \mathbf{a}_j \in \mathbb{R}^d$ , leur synergie peut être définie par une **distance inversée** ou une **similitude**, telle que

$$S(\mathcal{E}_i, \mathcal{E}_j) = \exp(-\gamma \|\mathbf{a}_i - \mathbf{a}_j\|^2),$$

pour un certain  $\gamma > 0$ . La **pondération**  $\omega_{i,j}(t)$  qui relie les entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  se met alors à jour selon la règle adaptative (voir section 1.4.5). Lorsque deux **fenêtres** ou segments audio affichent une forte similarité (timbre commun, bruit de fond identique, même locuteur), leurs liaisons  $\omega_{i,j}$  se **renforcent**, initiant ainsi la création d'un **cluster** rassemblant ces entités, et permettant de **repérer** un même motif acoustique. Cette démarche d'**auto-organisation** remplace l'approche classique qui découpe l'audio selon des règles fixes : le réseau laisse les liaisons décider d'une segmentation ou d'une agrégation adaptative.

De plus, si un **environnement** se modifie (bruit important, micro défaillant, accent inhabituel), des entités précédemment utiles peuvent perdre leur **synergie** avec les autres composantes, faisant décroître  $\omega_{i,j}$ . L'entité se retrouve alors **isolée** et n'influence plus le cœur du réseau. Inversement, lorsqu'une nouvelle configuration sonore émerge (baisse du bruit, nouveaux sons caractéristiques), d'autres **connexions**  $\omega$  apparaissent ou se renforcent, configurant de nouveaux **clusters** sonores. Ainsi, la **reconnaissance** de motifs acoustiques (sirènes, voix enfantines, grondements) se met en place de manière **continue**, sans qu'on doive réentraîner de bout en bout un réseau rigide. Le **DSL** s'adapte localement, confortant l'idée d'un système plus **résilient** et **évolutif** dans le traitement de l'audio.

### 1.6.2.2. Traitement du Langage Naturel : Entités Lexicales et Sémantiques

Dans les méthodes **classiques** de **traitement du langage naturel** (NLP), des modèles (RNN, LSTM, Transformers) s'appuient sur des **embeddings** (word2vec, GloVe, BERT, etc.) et des mécanismes d'attention ou de convolution. Bien qu'ils aient atteint d'excellents résultats (traduction, question-réponse, résumé, etc.), ils s'avèrent fortement dépendants d'un **entraînement** supervisé ou auto-supervisé de grande ampleur, et peinent à **intégrer** de la **logique** ou du **symbolique** sans recourir à des mécanismes additionnels. De plus, la **structure** de ces modèles demeure souvent linéaire (ou en arbre pour l'analyse syntaxique), imposant le traitement séquentiel des tokens.

Le **Deep Synergy Learning (DSL)** propose une alternative plus **flexible**. Au lieu d'imposer une séquence rigide de tokens à travers un pipeline (embedding + attention + couches finales), on peut **déployer** un **réseau** d'entités couvrant les différentes dimensions du texte. Chaque entité  $\mathcal{E}_i$  peut représenter un **mot** (ou token) muni d'un embedding  $\mathbf{w}_i$ , un **nœud syntaxique** extrait d'un parse tree (rôle de sujet, verbe, complément), ou un **concept sémantique** (topic, entité nommée, etc.). Ces entités s'**auto-organisent** si leur synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$  s'avère suffisamment élevée pour justifier un **regroupement**.

On peut alors voir naître des **clusters** lexical-sémantique regroupant des mots liés à un même champ lexical, un **verbe** et ses **compléments**, ou plusieurs **concepts** partagés par diverses phrases. Contrairement à une **arborescence** imposée (comme dans une analyse syntaxique standard), le **SCN** peut se montrer plus **fluide** : si un mot possède plusieurs sens (polysémie), la synergie qu'il entretient avec certains topics oriente la sélection de celui-ci, ce qui peut désambiguïser l'interprétation. De même, des liaisons  $\omega_{i,j}$  peuvent se **rompre** si un ensemble de mots cesse d'être pertinent dans un nouveau contexte textuel. L'**auto-organisation** rend donc la structure du **réseau** ajustable, permettant d'incorporer ou de dissoudre des sous-groupes de tokens ou de concepts selon qu'un **gain** mutuel se manifeste, ouvrant ainsi la voie à une **compréhension** plus modulable du langage.

### 1.6.2.3. Couplage Audio-Texte et Auto-Fusion Synergiques

De nombreux systèmes doivent traiter simultanément un **flux audio** (voix, intonations, bruits) et un **texte** associé (transcriptions, sous-titres). Le **Deep Synergy Learning (DSL)** s'avère particulièrement bien adapté à ce scénario, car il ne fixe pas a priori la manière de "joindre" l'audio et le texte. Au lieu d'un alignement imposé par un algorithme dédié, on laisse le **Synergistic Connection Network (SCN)** réguler les liaisons  $\omega$  entre des entités  $\mathcal{E}_{\text{aud}}$  et des entités  $\mathcal{E}_{\text{txt}}$ . Si la **co-information** ou une autre mesure de **similarité** suggère qu'un certain segment sonore correspond à un certain segment textuel, la pondération  $\omega_{\text{aud},\text{txt}}$  se renforce, conduisant à la création d'un **cluster** multimodal.

Parmi les entités audio, on peut distinguer des segments temporels ou des frames associées à une **fenêtre** spectrale spécifique. Du côté textuel, on dispose d'entités représentant des **mots**, des **tokens**, voire des syntagmes plus longs. La mise à jour des liens  $\omega_{\text{aud},\text{txt}}$  procède alors par la règle adaptative (section 1.4.5), détectant les **paires** qui coïncident fréquemment. Ainsi, lorsqu'un segment audio s'aligne régulièrement sur une suite de mots, on observe un gain synergique. On forme donc un **cluster** englobant les parties orales et leurs correspondances textuelles, sans qu'un schéma d'alignement fixe soit nécessaire.

Un tel mécanisme confère une **robustesse** notable. Si des segments audio se montrent inintelligibles (forte perturbation, accent trop marqué), la **synergie** s'affaiblit, et les liaisons avec les entités textuelles cessent de croître, isolant ces segments pour ne pas polluer l'ensemble. Le réseau **accepte** ainsi l'hétérogénéité de la qualité sonore et s'auto-adapte. En outre, cette logique **permet de découvrir** des co-occurrences ou des associations inhabituelles, par exemple un mot (ou un motif lexical) qui se répète chaque fois qu'un type particulier de sonorité apparaît. Le **DSL** parvient donc à fusionner audio et texte de manière plus **flexible** que ne le ferait un pipeline d'alignement rigide, et il favorise l'intégration de nouvelles données ou la détection de nouveaux mots/sons sans devoir procéder à un réentraînement ou une refonte exhaustive du modèle.

#### 1.6.2.4. Évolutions Possibles et Défis Mathématiques

Dans l'application du **Deep Synergy Learning (DSL)** à l'audio, au texte ou à des mélanges de modalités, plusieurs difficultés et pistes d'extension apparaissent.

Un premier enjeu concerne la **complexité**. Comme en vision (section 1.6.1.4), on peut générer un grand nombre d'entités audio (fenêtres temporelles, frames) ou textuelles (tokens, syntagmes). Le **graph** qui en résulte peut ainsi atteindre une taille considérable, rendant la mise à jour des pondérations  $\omega_{i,j}$  et l'estimation des synergies coûteuses. Il est donc souvent nécessaire de recourir à des **stratégies** parcimonieuses pour limiter la prolifération de liens, comme échantillonner périodiquement les entités, fusionner les entités jugées redondantes ou fixer un seuil  $\omega_{\min}$  qui supprime les connexions trop faibles (section 1.4.5).

En second lieu, on notera que la **modélisation n-aire** peut se révéler essentielle. Les **synergies** binaires (audio–audio, texte–texte ou audio–texte) ne suffisent pas toujours à capturer des phénomènes impliquant trois (voire davantage) entités. Il se peut, par exemple, que l'union de deux fragments audio et d'un groupe de mots textuels produise une information inatteignable avec des paires isolées. La **synergie n-aire** (section 1.4.7) permet de refléter ces complémentarités plus complexes, au prix d'un accroissement de la difficulté mathématique, car il faut évaluer l'information ou la similarité simultanément sur plusieurs variables.

Enfin, la possibilité d'**hybridation symbolique** apparaît particulièrement intéressante dans le domaine du **NLP** avancé (raisonnement, question-réponse complexes). Comme évoqué en section 1.5.7, on peut introduire des **règles logiques** ou des **entités** symboliques, lesquelles interagissent avec les **entités textuelles**. La synergie se définit alors par une fonction reliant le sens des mots à des conditions logiques ou des assertions formelles. Si cette intégration accroît encore la complexité, elle apporte un surcroît de **capacité cognitive**, autorisant un raisonnement plus poussé et assurant une **explicabilité** renforcée, dans l'esprit d'une IA **neuro-symbolique** où la dynamique des pondérations répond à la fois aux critères sub-symboliques (similarité, co-information) et aux axiomes symboliques.

#### 1.6.2.5. Conclusion Partielle : un Cadre Évolutif pour l'Audio et le Langage

Le **Deep Synergy Learning (DSL)** s'affirme comme une **approche** ou un **complément** aux architectures traditionnelles (CNN, RNN, Transformers) dans l'analyse de **flux audio** et le **traitement du langage naturel**. Contrairement aux pipelines classiques, le **DSL** ne se fonde pas sur un découpage préalable ou un alignement imposé, mais laisse les entités acoustiques ou lexicales s'**auto-organiser** dès que leur synergie l'indique. Cette logique confère plusieurs **avantages** :

L'**auto-organisation** pilote la segmentation et l'alignement de manière adaptative : si certains segments audio apparaissent corrélés ou si certaines paires de mots se révèlent complémentaires, leurs pondérations respectives se consolident, même en l'absence d'un découpage figé. Le **réseau** n'a pas à imposer un ensemble de tranches temporelles ni une procédure d'alignement prédéterminée. Il s'en dégage une **robustesse** face aux sources de variations dans les flux audio (bruit, accents, modifications de la source) et face aux registres du langage (polysémie, style). En cas d'incertitude ou de qualité médiocre, les liens connexes s'affaiblissent sans qu'un réapprentissage général soit nécessaire.

La **fusion** audio–texte se trouve également facilitée : plutôt que d'exiger un couplage strict d'un segment sonore et de sa transcription, la dynamique interne du **DSL** découvre librement les correspondances, détectant les segments oraux et textuels qui coïncident dans le temps ou dans le sens. Cette méthode rend compte d'une **flexibilité** avancée, car le système peut absorber de nouveaux sons, un nouveau vocabulaire ou même des **règles** symboliques sans devoir être reconstruit. L'ajout d'entités inopinées dans le graphe suffit, et le **système** gère la mise à jour des pondérations.

Ces caractéristiques ouvrent la voie à plusieurs **applications** évolutives : la **reconnaissance vocale** en environnement bruité s'en trouve renforcée ; la **compréhension** de conversations multimodales (avec indices vidéo et audio) se gère sans pipeline d'alignement contraint ; la **traduction flexible** ou la **détection** d'anomalies audio-linguistiques profitent de la structure distribuée et dynamique. Le tout s'appuie sur les **règles d'auto-organisation** (pondérations adaptatives, constitution de clusters) et de **synergie** (distance, co-information, etc.), qui instaurent un fonctionnement moins tributaire d'un entraînement supervisé exhaustif et davantage tourné vers l'évolution continue et la **coopération** locale entre entités.

### 1.6.3. Robotique et Systèmes Intelligents Évolutifs

Les concepts d'**auto-organisation**, de **synergie** et de **coopération** entre entités, propres au **Deep Synergy Learning (DSL)**, prennent tout leur sens dans le domaine de la **robotique** et des **systèmes intelligents**. En effet, la robotique moderne se caractérise par :

- Une **intégration** de multiples capteurs (vision, audio, pression, position, etc.),
- Des **actions** diverses (mouvements, manipulations, interactions) à coordonner dans des environnements changeants,
- La nécessité d'une **adaptation continue** (à la dynamique physique, aux aléas de l'environnement),
- Des **défis** de planification et de décision temps réel, souvent avec de l'**incertitude** et du **bruit**.

Cette sous-section (1.6.3) montre comment le **DSL**, grâce à ses mécanismes de **pondérations adaptatives** et à ses **clusters auto-organisés**, peut être particulièrement utile à la **robotique** et aux **systèmes intelligents**. Elle décrit :

15. L'**approche distribuée** que le DSL propose pour les robots multi-capteurs,
16. La **coévolution** des entités sensorielles et motrices,
17. La capacité à **s'auto-réorganiser** lorsqu'un robot ou un système intelligent découvre de nouveaux modules (nouveaux effecteurs, nouveaux contextes, etc.),
18. Des exemples d'applications, depuis les **robots collaborateurs** jusqu'aux **systèmes autonomes** en changement permanent.

#### 1.6.3.1. Robotique Multi-Capteurs : une Structure Synergique

Dans un contexte robotique, un système typique réunit plusieurs **capteurs** disposés sur la plateforme, pouvant inclure une **caméra** (analyse visuelle), un **LIDAR** ou un **radar** (cartographie de l'environnement), des **capteurs de pression** ou de **toucher**, des **capteurs inertiels** (IMU), un **microphone** pour la partie audio, etc. Les approches traditionnelles (notamment basées sur ROS et des architectures préconçues) tendent à définir un **pipeline** de fusion, auquel chaque capteur envoie ses données, puis une **couche** de décision. Cette rigidité suppose un schéma de liaison explicite entre capteurs et modules supérieurs, imposant par exemple un module de traitement conjoint image-LIDAR si l'on veut superposer un nuage de points laser et un flux de caméra.

Le **Deep Synergy Learning (DSL)**, quant à lui, permet d'instituer un **ensemble** d'entités  $\{\mathcal{E}_{cam}, \mathcal{E}_{lidar}, \dots\}$  correspondant chacune à une source sensorielle ou à un bloc fonctionnel plus abstrait. Chacune de ces entités peut juger de la **synergie** qu'elle partage avec les autres : si la caméra  $\mathcal{E}_{cam}$  et le LIDAR  $\mathcal{E}_{lidar}$  constatent régulièrement la détection d'un même obstacle, la valeur associée à leur liaison se consolide, menant à un **renforcement** de  $\omega_{cam,lidar}$ . Inversement, lorsqu'un capteur inertiel s'avère temporairement *bruité* (information incohérente par rapport aux autres lectures), ses connexions décroissent naturellement jusqu'à s'affaiblir et, le cas échéant, s'isoler.

Grâce à ce principe d'**auto-organisation**, il devient possible qu'un **cluster** multimodal émerge pour caractériser une situation telle qu'un "terrain accidenté" perçu simultanément par la caméra (textures complexes), le LIDAR (irrégularités de distance) et peut-être un micro détectant des bruits anormaux. Ce cluster se **dissout** ensuite lorsque le robot quitte la zone problématique ou que la configuration change. Le **réseau** d'entités agit donc de manière plus **dynamique** et **distribuée**, sans imposer un module de fusion central dédié à chaque couple de capteurs. Les liaisons  $\omega_{i,j}$  se forment ou se dissolvent localement, en fonction de la **valeur ajoutée** mesurée à travers la synergie entre entités, conférant au système une **résilience** plus élevée, qu'il s'agisse de gérer un capteur défaillant, l'arrivée d'un nouveau module, ou un changement inattendu dans l'environnement.

#### 1.6.3.2. Action, Effecteurs et Synergie Motrice

En robotique ou dans tout système intelligent disposant de **modules d'action** (moteurs, articulations, roues, pinces), on adopte généralement un paradigme où un **contrôleur** (PID, MDP, RL, etc.) reçoit un **état** en provenance des capteurs et génère des **commandes**. Le **Deep Synergy Learning (DSL)** aborde la question autrement, en considérant les effecteurs comme des **entités**  $\mathcal{E}_{motrice}$  au même titre que les capteurs. Par exemple, chacune des articulations d'un bras, ou la rotation d'une roue, ou la position d'une pince peut être dépeinte par un vecteur  $\mathbf{x}_{motrice}$ . La **coopération** avec les entités sensorielles s'établit dès lors que la **synergie**  $S(\mathcal{E}_{capteur}, \mathcal{E}_{motrice})$  s'avère positive, incitant à **renforcer** la pondération  $\omega_{capteur, motrice}$ .

Un bras articulé qui coordonne son mouvement avec un flux visuel voit, par exemple, la pondération  $\omega_{cam, bras}$  croître si la caméra confirme que la pince se trouve dans la bonne position. À l'inverse, un joint dysfonctionnel ou mal calibré cesse d'apporter un réel gain, ce qui fait chuter la **synergie**, rendant son influence moindre dans la stratégie globale. Le **SCN** assure ainsi une forme d'**adaptation** où les effecteurs pertinents coopèrent étroitement avec les capteurs produisant des informations utiles, sans qu'il soit nécessaire de configurer explicitement un contrôleur central devant orchestrer toutes les modalités.



On peut également insérer un **signal** de récompense ou un indicateur de performance  $R$  qui module la synergie. Si la coopération entre un capteur  $\mathcal{E}_{\text{capteurA}}$  et un effecteur  $\mathcal{E}_{\text{motriceB}}$  accroît ce score (par exemple, la manipulation d'un objet est mieux réalisée, ou la trajectoire est plus stable), la **pondération**  $\omega_{\text{capteurA}, \text{motriceB}}$  s'en trouve augmentée. Cette logique correspond à un **processus** de “learning by synergy”, sans avoir à recourir à une formalisation en MDP (Markov Decision Process) rigide. Les entités effectrices et sensorielles **s'auto-organisent** localement via la mise à jour des connexions  $\omega$ , découvrant progressivement quelles combinaisons capteur–action se révèlent fructueuses pour atteindre l'objectif.

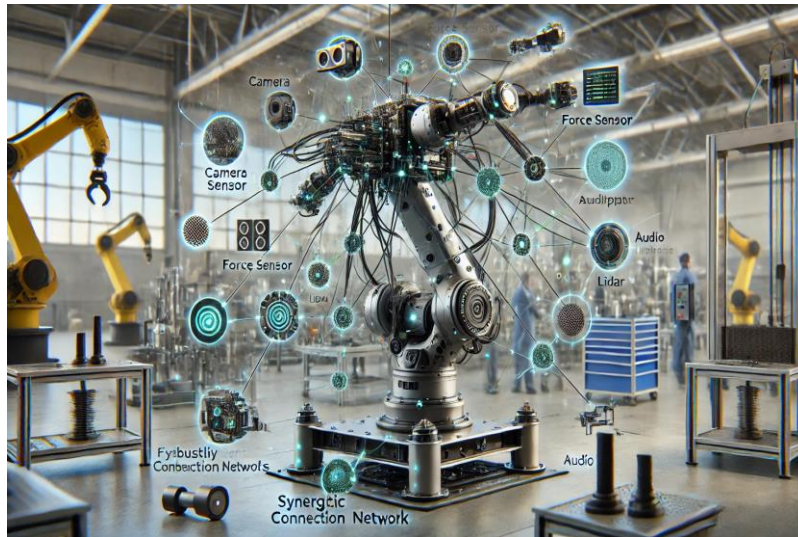
### 1.6.3.3. Adaptation Continue et Évolution Modulaire

Un des atouts fondamentaux du **Deep Synergy Learning (DSL)** tient dans sa **capacité d'évolution** : lorsqu'un robot se voit doté d'un nouveau **capteur**, d'un **nouvel effecteur** (par exemple un autre bras), ou que l'on retire un module obsolète, on se contente d'**ajouter** (ou de **supprimer**) l'entité correspondante  $\mathcal{E}_{\text{nouveau}}$  au sein du **Synergistic Connection Network (SCN)**. Les pondérations  $\omega_{\text{nouveau}, \cdot}$  suivent la dynamique habituelle (voir section 1.4.5) et se **renforcent** ou se **dissolvent** en fonction de la **synergie** détectée. Il n'est donc pas requis de recompiler l'ensemble du pipeline ou de procéder à un **réapprentissage** complet d'un grand réseau neuronal. La **logique** adaptative se borne à évaluer localement la valeur ajoutée de la nouvelle entité, puis à l'intégrer dans un **cluster** pertinent si besoin.

Cette **propriété** ne se limite pas à la **robotique** physique. Dans des **systèmes logiciels** complexes, par exemple dans l'administration de centres de données ou d'architectures distribuées, le principe reste valable. Chaque microservice, base de données, module de monitoring, etc., peut être considéré comme une **entité**  $\mathcal{E}_{\text{service}}$ . L'arrivée d'un nouveau service, la suppression (ou la panne) d'un service existant, se modélise par l'**ajout** ou le **retrait** d'entités correspondantes, et les **liaisons**  $\omega$  se régulent selon la **coopération** réelle (échange d'informations, dépendances). Les **clusters** qui en résultent rassemblent les **services** ayant des affinités ou co-occurrences fortes, autorisant une **répartition** et une **coordination** modulaires.

Ce **principe d'auto-organisation** évite la centralisation rigide d'un pipeline complet ou la révision constante d'un réseau monolithique. Au contraire, le **SCN** se reconfigure **spontanément** : il repère quels éléments demeurent fiables et utiles, et il restructure graduellement l'architecture face à de nouveaux besoins ou de nouvelles entités. Cette démarche s'inscrit dans l'esprit du **lifelong learning**, où le **réseau** lui-même façonne en continu sa **topologie** et maintient un équilibre entre la conservation de connaissances antérieures et l'ajout de briques inédites.

### 1.6.3.4. Illustration : Robot Collaboratif Multicapteurs



Un **robot collaboratif** (cobot) dans un environnement industriel peut disposer de multiples capteurs et actionneurs. Par exemple, on considère une **caméra** pour localiser la pièce à assembler, un **capteur de force** dans la pince pour jauger la préhension, un **LIDAR** détectant la distance aux opérateurs humains pour prévenir les collisions, un **module audio** apte à recevoir des instructions vocales, et enfin les actionneurs : la **pince**, le **bras articulé** et un **moteur** de déplacement sur plateforme.

Au sein d'un **Synergistic Connection Network (SCN)** selon les principes du **Deep Synergy Learning (DSL)**, chaque composant devient une **entité** :  $\mathcal{E}_{\text{cam}}$ ,  $\mathcal{E}_{\text{force}}$ ,  $\mathcal{E}_{\text{lidar}}$ ,  $\mathcal{E}_{\text{audio}}$  pour les capteurs, et  $\mathcal{E}_{\text{pince}}$ ,  $\mathcal{E}_{\text{bras}}$ ,  $\mathcal{E}_{\text{moteur}}$  pour les effecteurs. Chacune évalue localement sa **synergie** avec les autres, de sorte que les liens  $\omega_{i,j}$  se **renforcent** s'il apparaît un réel bénéfice à leur coopération. Lorsque la caméra coopère régulièrement avec la pince — par exemple, parce que la zone repérée par l'image correspond effectivement à la position où la pince intervient —, la liaison  $\omega_{\text{cam}, \text{pince}}$  se consolide, engendrant un **cluster** local regroupant ces entités {cam, pince}. Si, au contraire, l'audio se

révèle trop bruité, son synergie avec le reste du système diminue et  $\omega_{\text{audio}}$  tombe en deçà du seuil de pertinence ; l'entité audio se retrouve alors **isolée** ou peu contributive jusqu'à ce que la qualité du signal s'améliore.

On peut intégrer un **signal de récompense** afin de consolider les combinaisons sensorielles et motrices les plus efficaces (par exemple, si le cobot réussit à saisir et assembler la pièce sans heurter de zones sensibles, on valorise cette configuration). Les entités impliquées voient leurs pondérations internes augmenter, ce qui renforce la probabilité de réutiliser ce mode de coopération. L'ensemble constitue un **réseau** qui s'**auto-adapte**, insérant ou affaiblissant des liens selon l'utilité détectée pour la tâche courante. Cette logique évite de reconstruire un pipeline complet ou de réentraîner un vaste réseau en cas de modifications (ajout d'un nouveau capteur, panne de l'actuel, variation des conditions d'assemblage). Les **clusters** multimodaux (vision + pince + capteur de force, par exemple) forment des **sous-réseaux** cohérents, naturellement conçus pour la tâche à effectuer.

### 1.6.3.5. Défis et Perspectives de Recherche

Lorsqu'on applique le **Deep Synergy Learning (DSL)** à la robotique multi-capteurs et multi-effecteurs, plusieurs difficultés et voies d'extension émergent, tant sur le plan pratique que théorique.

Un premier défi concerne la **complexité combinatoire**. Dans un robot doté de multiples capteurs (caméra, LIDAR, IMU, microphone...) et d'autant de modules d'action (divers joints, roues, pinces), le **graphe** des pondérations  $\omega$  peut rapidement croître en taille, rendant la mise à jour de chaque lien difficile à maintenir en **temps réel**. Des stratégies de **parsimonie** (seuils éliminant les liaisons trop faibles) ou des mises à jour strictement **locales** (limiter le calcul aux entités concernées par une nouvelle donnée) se révèlent indispensables pour garantir la **scalabilité**.

Un deuxième point aborde les **systèmes multi-robots**. Lorsqu'une flotte (drones, véhicules, robots industriels) doit coopérer, on peut concevoir un **réseau synergique** à plus grande échelle, où chaque robot forme un **sous-graphe** d'entités sensorielles et motrices. Les **synergies** inter-robots introduisent alors des liaisons  $\omega$  entre entités de différents appareils. Il devient possible d'imaginer une **co-organisation** sur l'ensemble de la flotte, favorisant la dynamique d'émergence de clusters inter-robots dès qu'un gain mutuel apparaît (partage de capteurs, répartition de tâches). L'envergure de ce problème soulève d'importants défis de **communication** et de gestion **distribuée** des liens, car on ne peut pas nécessairement centraliser toutes les informations.

La **sécurité** et la **robustesse** forment également un axe de recherche majeur. En robotique, il est impératif de respecter des contraintes physiques (éviter la collision, maintenir une distance de sécurité). Dans le **DSL**, on peut introduire des pénalités ou des **contraintes** pour réduire la pondération  $\omega_{i,j}$  lorsqu'une configuration s'avère dangereuse ou enfreint des invariants mécaniques ou de sûreté. De cette manière, l'**auto-organisation** se soumet à des garde-fous, garantissant un comportement sûr même lorsque la structure du réseau évolue.

Enfin, l'idée d'**hybridation symbolique** (section 1.5.7) se transpose à la robotique avancée et aux tâches haut niveau. Les entités symboliques, représentant des **règles** ou des **plans**, peuvent interagir avec les entités sensorielles et motrices : la synergie entre la **logique** (définissant un plan d'action) et la **perception** (reconnaissance visuelle, retour de force) oriente la réorganisation du graphe, débouchant sur une planification distribuée et **cognitive**, où la coopération se fait au croisement du symbolique (ordre de mission) et du sub-symbolique (capteurs, effecteurs). L'ensemble de ces pistes ouvre un **vaste** espace de recherche pour faire évoluer la robotique vers des systèmes de décision auto-organisés, hautement modulaires et résilients.

## Conclusion

Dans la **robotique** et les **systèmes intelligents évolutifs**, le **Deep Synergy Learning** :

- Offre une **approche distribuée** pour intégrer de multiples capteurs et effecteurs,
- Gère la **reconfiguration** continue (bruit, arrivée de nouveaux modules, pannes),
- Favorise l'**apprentissage** local, la formation de **clusters** sensorimoteurs stables,
- Permet un fonctionnement **plus organique** que le pipeline rigide (capteur → fusion → décision), et plus extensible face à l'évolution de l'architecture robotique.

Cette capacité d'**auto-organisation** s'avère essentielle lorsque le robot (ou le système) doit s'ajuster rapidement, adopter de nouveaux modules ou se coordonner avec d'autres entités (robots, agents, microservices). On retrouve ainsi la philosophie déjà mise en avant dans le DSL : éviter la hiérarchie monolithique et le réapprentissage complet, au profit d'une **dynamique** adaptative, guidée par la **synergie** entre entités.



#### 1.6.4. Recommandation Personnalisée et Systèmes de Décision

Au-delà de la robotique (1.6.3), le **Deep Synergy Learning (DSL)** trouve également une place de choix dans les **systèmes de recommandation** et plus largement les **systèmes de décision**. Dans ces domaines, il s'agit souvent de traiter des **profils utilisateurs**, des **contenus** (produits, articles, films, musiques) ou des **options** (plans d'action, configurations possibles), pour aboutir à des **conseils** ou des **choix** pertinents. Les approches traditionnelles (collaborative filtering, réseaux neuronaux de recommandation, arbres de décision) présentent généralement des structures fixes : on définit un modèle qui, après un entraînement plus ou moins supervisé, calcule des scores de préférence.

Le **DSL**, par sa **dynamique d'auto-organisation** et sa **prise en charge** des liens synergiques, propose une voie plus **flexible** et **adaptative**. Il peut gérer à la fois :

- Des **entités “utilisateurs”**  $\{\mathcal{E}_u\}$  représentant différentes personnes (ou profils),
- Des **entités “contenus”**  $\{\mathcal{E}_c\}$  décrivant articles, produits, médias, etc.,
- Des **entités “contextes”** (saisons, heures, tendances récentes),
- Éventuellement des **entités symboliques** (règles de business logic, préférences explicites) si on veut un système cognitif plus avancé.

Nous allons voir en détail comment ces entités interagissent et comment le DSL peut aider à **adapter** les recommandations au fil du temps, en favorisant les **clusters** (ou micro-réseaux) les plus cohérents, et en révisant les connexions obsolètes.

##### 1.6.4.1. Entités Utilisateurs et Contenus : un Graphe Synergique

Les **systèmes de recommandation** s'appuient classiquement sur une **matrice  $\mathbf{M}$**  dont les lignes représentent les utilisateurs  $\{u \in U\}$  et les colonnes les contenus  $\{c \in C\}$ . Chaque entrée  $\mathbf{M}_{u,c}$  correspond à l'intérêt ou la note attribuée par l'utilisateur  $u$  au contenu  $c$ . Les approches courantes (collaborative filtering, factorisation matricielle, autoencodeurs de complétion...) consistent alors à factoriser  **$\mathbf{M}$**  ou à prédire les valeurs manquantes par un entraînement supervisé ou semi-supervisé. Cette vision demeure matricielle : on y ajoute rarement d'autres entités hors du couple (utilisateurs, contenus).

Dans le **Deep Synergy Learning (DSL)**, on substitue ou complète cette approche en définissant un **graphe** évolutif. On introduit d'abord un ensemble d'**entités**  $\{\mathcal{E}_u\}_{u \in U}$  représentant les utilisateurs et un autre  $\{\mathcal{E}_c\}_{c \in C}$  pour les contenus (articles, films, produits, etc.). Si nécessaire, on y ajoute aussi des **entités contextuelles**  $\{\mathcal{E}_{\text{time}}, \mathcal{E}_{\text{location}}, \dots\}$  lorsqu'on souhaite prendre en considération la dimension temporelle ou spatiale. Les **liaisons**  $\omega_{u,c}(t)$  reliant un utilisateur  $\mathcal{E}_u$  à un contenu  $\mathcal{E}_c$  sont mises à jour suivant une **règle adaptative** (voir section 1.4.5) : elles se renforcent si la “synergie”  $S(\mathcal{E}_u, \mathcal{E}_c)$  est jugée suffisamment positive, ou au contraire déclinent si aucune utilité mutuelle n'est perçue. La fonction  $S$  peut reposer sur une **similarité** d'intérêts, une **co-information** d'usage, ou sur l'idée d'un gain mesurable (par ex. le nombre de clics ou de vues).

Grâce à ce schéma, on assiste à la formation spontanée de **clusters** regroupant simultanément des utilisateurs et des contenus. Un **cluster** peut ainsi rassembler plusieurs amateurs de rock, divers albums rock et, s'il existe des entités de contexte, certaines soirées “concert” associées, révélant une forte **co-occurrence** ou un alignement d'intérêts. Comparé à la matrice  **$\mathbf{M}$**  — qui se borne aux couples  $(u, c)$  — le **SCN** tolère l'ajout de toute entité auxiliaire. Il devient possible de **fusionner** des groupes si une synergie apparaît entre leurs membres, ou d'**isoler** un contenu dont l'intérêt décroît soudain auprès des utilisateurs concernés. En outre, l'arrivée d'un nouvel utilisateur  $\mathcal{E}_{\text{nouveau}}$  ou d'un nouveau contenu  $\mathcal{E}_{\text{item}}$  se gère en insérant simplement l'entité, puis en laissant la synergie ajuster les liens  $\omega_{\text{nouveau}, \cdot}$ . Le réseau se **restructure** localement, sans imposer une refonte globale comme le ferait une ré-analyse complète de la matrice  **$\mathbf{M}$** . Les sections suivantes (voir 1.6.4.2 et au-delà) approfondiront comment cette logique d'auto-organisation enrichit la **recommandation** par la capacité à intégrer de nouvelles entités et à effectuer une fusion dynamique des préférences et des contenus.

##### 1.6.4.2. Mise à Jour Adaptative et Évolution dans le Temps

Dans un **système de recommandation**, la **distribution** se modifie sans cesse : il apparaît de nouveaux utilisateurs, de nouveaux contenus (articles, livres, films récents), des préférences se réorientent au fil des modes, des promotions ou des saisons. Les algorithmes classiques de filtering (factorisation, deep learning supervisé) doivent alors réestimer leurs paramètres, au risque d'exiger une nouvelle phase de réapprentissage ou un raffinement partiel, avec le danger de dérégler l'équilibre déjà atteint.

Le **Deep Synergy Learning (DSL)** adopte une démarche plus organique. Lorsque des entités inédites  $\mathcal{E}_{\text{nouveauUser}}$  ou  $\mathcal{E}_{\text{nouveauContenu}}$  apparaissent, on les **insère** directement dans le **Synergistic Connection Network**. Le mécanisme de mise à jour  $\omega_{i,j}$  (section 1.4.5) se charge alors de **consolider** ou non les liaisons selon la **synergie** qu'elles dégagent. Ainsi, un nouveau contenu sera rapidement lié (pondérations positives) aux utilisateurs qui partagent des goûts proches ou qui interagissent avec ce contenu, conduisant à la formation

d'un **cluster**. À l'inverse, un contenu ancien qui perd de l'intérêt voit ses pondérations  $\omega_{u,\text{contenu}}$  décroître naturellement, jusqu'à s'isoler si plus personne ne s'y intéresse.

De plus, le **DSL** n'est pas cantonné aux seules entités "utilisateur" ou "contenu". Des **entités contexte** comme la **saison** ( $\mathcal{E}_{\text{été}}$ ), la **localisation** ( $\mathcal{E}_{\text{Paris}}$ ), un **événement** ( $\mathcal{E}_{\text{Noël}}$ ) peuvent influencer la **synergie**. Si la proximité entre un événement (Noël), certains contenus (articles cadeaux, recettes festives) et un profil d'utilisateurs se concrétise, alors les liens  $\omega_{\text{Noël}, \text{contenu}}$  et  $\omega_{\text{Noël}, \text{utilisateur}}$  s'amplifient, façonnant un **cluster** "Noël + X utilisateurs + Y contenus". Lorsque la période de Noël s'achève, ces mêmes liaisons perdent peu à peu de leur attrait et se **délient**, laissant place à d'autres configurations contextuelles (telles que "soldes de janvier"). Le réseau se **reconfigure** ainsi localement, sans imposer un réentraînement massif ni une réinitialisation de tout le modèle, et garantit une adaptation continue au gré des changements de **distribution**.

#### 1.6.4.3. Décision et Recommandation Basées sur les Pondérations Synergiques

Une fois le **réseau** de synergie  $\{\omega_{i,j}\}$  formé ou relativement stabilisé (au moins à un instant donné), il devient possible d'en **extraire** un score de recommandation pour un **utilisateur**  $\mathcal{E}_u$  et un **contenu**  $\mathcal{E}_c$ . Une formule concevable, inspirée d'une multiplication matricielle partielle, consiste à sommer l'**influence** indirecte par d'autres entités  $\{\mathcal{E}_k\}$  (utilisateurs proches, tags contextuels, règles). On peut ainsi écrire :

$$\text{score}_{u,c} = \sum_k \omega_{u,k} \omega_{k,c}.$$

Intuitivement, si l'utilisateur  $\mathcal{E}_u$  et un contenu  $\mathcal{E}_c$  ne partagent pas de lien direct, on évalue leur **affinité** en passant par des entités  $\mathcal{E}_k$  qui peuvent être d'autres **utilisateurs** (similaires) ou des **concepts** (genre musical, catégorie d'article). Cet agrégat de contributions rappelle la structure de  $(W^2)_{u,c}$  dans le cadre d'une multiplication de la matrice des pondérations  $W$  par elle-même, sans s'y limiter nécessairement : des **variantes** plus subtiles peuvent être introduites pour mieux prendre en compte les chemins les plus pertinents ou un **cluster** local.

Lorsqu'on inclut des **entités symboliques** (section 1.5.7), portant des **règles** ou **contraintes** métier (par exemple "Ne pas recommander les contenus PG-13 à un compte junior"), on peut autoriser la **synergie** à se moduler en conséquence : si la liaison  $\omega_{\text{rule}, \text{contenu}}$  indique une incompatibilité (règle violée), elle réduit la pondération globale  $\omega_{\text{user}, \text{contenu}}$ . Il en résulte un **réseau** qui intègre à la fois les **préférences** non supervisées (co-occurrences utilisateurs/contenus) et des **limites** ou **politique** explicites (règles symboliques). Le **SCN** en tient compte localement, ce qui équivaut à un **filtrage** ou à un **re-rank** dynamique, plus flexible qu'un pipeline figé. La **recommandation** se rapproche alors d'une conclusion collective de l'ensemble des entités, chaque liaison  $\omega$  traduisant la **valeur** ou la **compatibilité** mutuelle.

#### 1.6.4.4. Illustrations Concrètes

Plusieurs exemples concrets permettent de mieux percevoir la **souplesse** et la **richesse** du **Deep Synergy Learning (DSL)** appliqué au domaine des systèmes de recommandation ou, plus largement, à l'**aide à la décision** sur des ensembles utilisateur-contenu-contexte.

Dans une **plateforme e-commerce**, on dispose d'entités représentant les **utilisateurs** (avec leur historique et profil), les **produits** (catégories, marques, etc.), et d'éventuelles entités de **contexte** (temps fort comme soldes, promotions, événements saisonniers). Au lieu de programmer explicitement des algorithmes de segmentation, le **DSL** laisse ces entités s'**agréger** en **clusters** si elles constatent un **gain** mutuel. Un sous-groupe d'utilisateurs peut ainsi s'avérer très intéressé par des produits d'une certaine marque ou style, formant un **cluster** qui se renforce dès lors que la synergie (p. ex. co-occurrence d'achats) s'avère constante. Les thématiques détectées évoluent naturellement en fonction des tendances ou de l'arrivée de nouvelles gammes de produits.

Dans un **service de streaming** (musique, vidéo), des entités peuvent correspondre aux **albums**, aux **artistes**, aux **playlists**, aux **tags** de genres (rock, pop, jazz), ainsi qu'aux **communautés** d'auditeurs. Le mécanisme de mise à jour  $\omega_{i,j}$  valorise la proximité entre un auditeur et un certain artiste, ou entre un artiste et un tag de genre, etc. Un **cluster** cohérent peut alors émerger autour de l'auditoire d'une tendance musicale particulière (rock), intégrant les artistes, les playlists populaires associées et les auditeurs fidèles. Si un nouvel artiste rejoint la plateforme, on ajoute une entité  $\mathcal{E}_{\text{artisteNew}}$  ; si celle-ci partage des caractéristiques avec le cluster rock (timbre de voix, style, tags), la **synergie** se trouve élevée, et l'artiste s'intègre au groupe. Les **recommandations** pour un auditeur donné s'en déduisent en analysant  $\omega_{x, \text{artiste}}$  ou bien  $(W^2)_{x, \text{artiste}}$  (cf. section 1.6.4.3), reflétant l'influence indirecte par d'autres auditeurs ou tags.

Dans un **système d'aide à la décision** industriel, on peut imaginer des entités décrivant différents **scénarios** (configurations de machines, paramètres de production), des **résultats passés** (qualité, temps de cycle), ainsi que des entités relatives aux **coûts** ou aux **ressources** (personnel, stocks). Les **liaisons**  $\omega_{i,j}$  se renforcent lorsqu'une **combinaison** de paramètres s'avère fructueuse pour la production, ou que deux scénarios se révèlent similaires et conduisent à un résultat positif. Le **DSL** élabore progressivement une **cartographie** flexible de l'espace décisionnel : de nouveaux scénarios peuvent être ajoutés sans réapprentissage global, et l'**auto-organisation** s'adapte si les conditions de production évoluent (matières premières, cadence, etc.). Le tout offre une vision plus **distribuée** et **réactive** que des approches rigides, car les entités ne cessent de mettre à jour leurs liaisons pour refléter la **synergie** détectée.

#### 1.6.4.5. Avantages et Défis

Dans un **système de recommandation** traditionnel, la construction d'un **pipeline** rigide est largement répandue, comme le collaborative filtering avec factorisation de matrice ; on effectue alors une estimation statique ou périodique, puis on l'emploie pour recommander. Le **Deep Synergy Learning (DSL)** propose au contraire une **auto-organisation** plus dynamique : lorsque les préférences d'un utilisateur fluctuent, ou lorsqu'un contenu quitte la base, la mise à jour des connexions  $\omega$  s'effectue localement, évitant un recalcul global. Cette **souplesse** améliore la réactivité à des évolutions telles que les promotions, la saisonnalité ou l'arrivée de nouveaux items.

On gagne aussi une **robustesse** et une certaine forme de **parsimonie** : des entités isolées (utilisateurs inactifs, contenus devenus obsolètes) perdent naturellement leurs liens  $\omega$  si leur synergie demeure trop faible. Elles n'interfèrent plus avec les autres recommandations, réduisant ainsi le **bruit**. Néanmoins, un **défi** majeur se manifeste si le nombre d'entités monte en flèche : il faut mettre en place des **heuristiques** (sparsification, échantillonnage, etc.) pour maintenir la faisabilité en grand volume (big data). Autrement, la multiplication des liaisons pourrait saturer les ressources de calcul.

Le **DSL** autorise aussi une **évolution contextuelle** : on peut introduire une nouvelle entité "festival d'automne", "temps pluvieux" ou "Black Friday" pour moduler en temps réel la synergie. Les liens  $\omega$  indiquent alors quels utilisateurs et quels contenus sont activés par ce contexte, sans qu'un réapprentissage massif soit nécessaire. Cela favorise une adaptation **continue** au gré des événements.

En dernier lieu, l'**explicabilité** s'en voit renforcée. Au lieu de recourir à des facteurs latents obscurs ( $F1, F2, \dots$ ), un **cluster** émergent peut être **interprété** plus directement : il rassemble un groupe d'utilisateurs et un sous-ensemble précis de contenus. Dans un même sous-réseau, l'entité "jazz" (tag de genre) et l'entité "club de blues" (contenu) peuvent se consolider avec un sous-groupe d'auditeurs, ce qui donne une **raison** explicite : "vous appartenez au cluster jazz-blues, d'où cette recommandation". Cette lisibilité dépasse la vision d'une factorisation matricielle standard, offrant un gain en transparence et facilitant la justification des suggestions.

#### Conclusion

Dans la **recommandation personnalisée** et les **systèmes de décision**, le **Deep Synergy Learning** se distingue par :

- Sa **capacité** à gérer une **multiplicité** d'entités (utilisateurs, contenus, contextes, règles),
- Son **auto-organisation continue**, autorisant l'ajout ou le retrait de modules sans réapprentissage global,
- Sa **formation de clusters** plus interprétables, où les associations (utilisateurs—contenus—contexte) émergent de la **synergie** plutôt que d'une factorisation imposée,
- Son **adaptation** en temps réel aux changements d'habitudes, de saison, de modes ou de distributions.

Avec le DSL, les **recommandations** et **décisions** se construisent dans un **réseau** évolutif, rendant possible une **cohérence** plus fluide avec des règles (symboliques), des signaux contextuels, ou des flux externes. C'est un champ où l'auto-organisation synergique peut apporter plus de **plasticité** et de **transparence** qu'un pipeline classique de filtering ou de scoring.

#### 1.6.5. Surveillance, Diagnostic Médical et Anomalies

Les principes d'**auto-organisation** et de **coopération adaptative** propres au **Deep Synergy Learning (DSL)** ne se cantonnent pas aux domaines précédemment cités (vision, audio, robotique, recommandation). Ils trouvent également une place toute particulière dans les **applications de surveillance**, de **diagnostic**, et de **détection d'anomalies**, qu'il s'agisse de contexte médical ou industriel. Dans ces environnements, on gère souvent d'importants **flux de données** en temps réel, provenant de **capteurs variés** (biologiques, physiques) qu'il faut **coordonner** pour repérer des situations anormales ou des symptômes précoces. Les systèmes classiques reposent sur des algorithmes d'**anomaly detection** (souvent supervisés, semi-supervisés, ou basés sur des seuils statiques) qui peuvent se révéler rigides ou trop dépendants d'un jeu de labels.

Le **DSL** introduit une **flexibilité** nouvelle : des **entités d'information** spécialisées dans différents signaux (capteurs, paramètres cliniques, logs machines), liées par des **pondérations synergiques** qui évoluent au fil du temps. Ainsi, l'**auto-organisation** permet de détecter des groupes d'observations cohérentes et, réciproquement, d'identifier des entités qui s'**écartent** de la synergie générale. Cette capacité de **repérage** local et de **réaction** dynamique rend le DSL particulièrement indiqué pour :

19. Les systèmes de **surveillance** (vidéo, capteurs environnementaux) qui doivent isoler des comportements suspects,
20. Les **diagnostics médicaux** complexes, où l'on fusionne imagerie, tests biologiques, relevés de signes vitaux pour repérer une pathologie,
21. L'**analyse d'anomalies** en maintenance prédictive, où un capteur ou un indicateur diverge du comportement habituel du cluster.

Nous allons voir plus en détail comment le DSL prend en compte la variété des signaux et repère, via la synergie (ou son absence), ce qui sort du lot et pourrait constituer une alerte ou un symptôme.

#### 1.6.5.1. Systèmes de Surveillance : Entités Multi-Capteurs, Scènes Vidéo

Dans de nombreux dispositifs de **surveillance vidéo**, les algorithmes classiques (souvent un pipeline à base de CNN et de suivi d'objets) doivent gérer un **flot** de données considérable et détecter en continu tout comportement ou événement anormal. Cette approche peut se révéler fragile face à la complexité des scènes ou à la diversité des conditions. Le **Deep Synergy Learning (DSL)** propose une alternative plus **auto-organisée** et **réactive**, en représentant chaque composante (segment visuel, contexte, schéma habituel) sous forme d'entités et en laissant la **synergie** orienter la structuration.

Dans le cadre d'une **vidéo** de surveillance, on peut définir des entités  $\mathcal{E}_{\text{seg},k}$  correspondant à différents **blobs** ou **segments** détectés dans la scène (régions de mouvement, silhouettes, etc.). D'autres entités  $\mathcal{E}_{\text{contexte}}$  décrivent, par exemple, l'heure, la zone surveillée, les règles de fréquentation habituelles. Enfin, des entités  $\mathcal{E}_{\text{pattern}}$  peuvent caractériser des **patrons** récurrents (le nombre moyen de personnes, la vitesse de déplacement habituelle, les régions autorisées ou interdites). Au sein du **Synergistic Connection Network**, les liaisons  $\omega$  se forment ou se réduisent selon la **cohérence** repérée : si un segment vidéo "blob A" s'inscrit habituellement dans la vitesse ou l'emplacement prévus, la synergie demeure haute. Lorsqu'un comportement s'écarte de l'ordinaire (un mouvement inhabituel, une densité de blobs anormale, etc.), la **pondération**  $\omega_{\text{seg}, \text{pattern}}$  chute, isolant le segment considéré et signalant une anomalie. L'algorithme peut alors émettre une alarme ou enclencher une inspection spécifique, sans nécessiter de recompiler ou de réentraîner un détecteur complet.

Dans la **surveillance** élargie, on ajoute souvent d'autres capteurs : mouvement (PIR), bruit, température, fumée, etc. Le **DSL** traite aisément cette hétérogénéité en insérant des **entités**  $\mathcal{E}_{\text{mouvement}}$ ,  $\mathcal{E}_{\text{son}}$ ,  $\mathcal{E}_{\text{température}}$ , dont les synergies s'**auto-régulent** avec les entités visuelles. Si un choc violent est détecté par  $\mathcal{E}_{\text{son}}$  alors que la vidéo ne rapporte aucun mouvement, la liaison  $\omega_{\text{son}, \text{cam}}$  diminue, suggérant un capteur défaillant ou un phénomène localisé sans trace visuelle. À l'inverse, la convergence de plusieurs capteurs (caméra détectant flammes, capteur son captant un "boom", etc.) renforce les liaisons, formant un **cluster** d'alerte qui intègre différentes sources. Cette **auto-organisation** rend la détection plus **résiliente** : les signaux douteux se voient mis à l'écart, tandis que les coïncidences réelles entre capteurs suscitent un regroupement cohérent, facilitant ainsi l'identification d'événements critiques.

#### 1.6.5.2. Diagnostic Médical : Entités Physiologiques et Multimodalité



Dans un **contexte médical**, les données proviennent de multiples **sources**. On dispose de l'**imagerie** (IRM, rayons X, scanner), de **tests biologiques** (formule sanguine, biochimie), de **signes vitaux** (ECG, pression, saturation en  $O_2$ ) et de **symptômes** rapportés, voire de données **génétiques** plus complexes. Un **réseau** de neurones classique, comme un CNN spécialisé dans l'interprétation d'IRM, a tendance à se focaliser sur un **type** de signal. Or, certains diagnostics difficiles (cancers rares, syndromes complexes) exigent un **croisement** entre diverses modalités, chacune partielle, parfois bruitée.

Le **Deep Synergy Learning (DSL)** se conçoit alors comme un **réseau** auto-organisé où des **entités**  $\{\mathcal{E}_{\text{irm}}, \mathcal{E}_{\text{analyseSang}}, \mathcal{E}_{\text{ecg}}, \dots\}$  représentent chacune un bloc de données ou un test particulier. Le **score** de synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$  indique si deux sources convergent vers une suspicion commune (par exemple, des anomalies visibles à l'IRM qui correspondent à des marqueurs sanguins inhabituels). Les **pondérations**  $\omega_{i,j}$

se renforcent lorsque la coopération de deux entités apporte un **gain** (détection corrélée), et un **cluster** pathologique peut se constituer, regroupant un sous-ensemble de paramètres (imagerie, analyses, symptômes) indiquant une même pathologie ou un syndrome précis.

De plus, l'évolution **temporelle** des données médicaux (phases aiguës ou rémissions) se gère de façon fluide : si un paramètre redevient normal, la synergie qu'il entretenait avec les marqueurs pathologiques s'épuise, et le **cluster** de la maladie se désolidarise progressivement de cette entité. Inversement, un nouveau symptôme peut s'y **agréger** dès qu'il présente une similarité ou co-occurrence notable avec d'autres facteurs pathologiques existants. Le **diagnostic** s'ajuste donc pas à pas, reflétant l'**état** du patient à chaque instant. On peut également insérer un **signal** de validation clinique ou de "récompense" lorsqu'un diagnostic s'avère correct, ce qui renforce les liens responsables de la bonne détection. Cette logique **auto-organisée**, distribuée et multimodale se distingue d'un pipeline rigide appliquant un classifieur sur l'imagerie ou l'analyse sanguine isolément : elle intègre **toutes** les sources, les pondérations  $\omega$  révélant la **coopération** la plus profitable à un moment donné.

### 1.6.5.3. Détection d'Anomalies Industrielles : Auto-Organisation et Isolement

En milieu **industriel** (production, logistique, transports), la **maintenance prédictive** implique une surveillance continue des capteurs (température, vibrations, courants moteurs, etc.) afin de détecter des signaux précurseurs de panne. Dans un **Deep Synergy Learning (DSL)**, on peut représenter chaque **capteur** par une entité  $\mathcal{E}_{\text{capteur}}$ , ajouter des entités décrivant des **normes** ou **règles** de fonctionnement (certaines pouvant être symboliques, voir section 1.5.7), ainsi que des **entités contextuelles** (charge de travail, température ambiante, etc.). Ces entités s'**auto-organisent** et forment un **cluster** principal (ou plusieurs) lorsqu'elles partagent un comportement cohérent.

Une **anomalie** se manifeste lorsque la **synergie**  $S(\mathcal{E}_k, \mathcal{E}_{\text{ref}})$  entre un capteur  $\mathcal{E}_k$  et le **cluster** de référence (normes, capteurs stables) chute. Contrairement à un système de seuils figés par capteur (par exemple "vibration > X" ou "température > Y"), le **DSL** confronte chaque capteur à l'ensemble de la structure en évolution : si un capteur dérive **seul** et ne suit plus la synergie habituelle, il se voit **isolé**, donc potentiellement en anomalie. À l'inverse, si plusieurs capteurs entament des dérives **cohérentes**, ils peuvent former un **sous-cluster** spécifique (un "cluster d'anomalie"), laissant deviner un défaut commun (ex. un roulement en fin de vie qui génère bruit et surchauffe).

Au fil du temps, un capteur fautif peut :

22. Se retrouver **isolé** dans un "espace d'anomalie" lorsque ses pondérations  $\omega_k$  décroissent,
23. Entrer en **coopération** avec d'autres entités également dérivantes, aboutissant à un nouveau **cluster** d'anomalie plus large,
24. Déclencher un **signal** de maintenance dès lors que la situation persiste ou si la synergie interne de l'anomalie (ex. bruit anormal + vibrations + température de roulement) continue de croître, signant une panne imminente.

Cette **auto-organisation** rend la détection plus adaptative : le **réseau** perçoit progressivement quelles **liaisons** demeurent fiables et lesquelles s'écartent. On évite ainsi les contraintes d'un pipeline rigide ou de seuils par capteur, car la logique de synergie estime la **relation** du capteur à l'ensemble du **cluster** de fonctionnement normal, autorisant un repérage de dérive **contextualisé** et évolutif.

### 1.6.5.4. Avantages et Défis : la Logique du DSL en Alerte et Diagnostic

Les applications de **Deep Synergy Learning (DSL)** dans le domaine de la **surveillance**, de la **maintenance** ou du **diagnostic** s'appuient sur sa capacité à constituer des **clusters** de manière autonome, en agrégeant des **entités** multiples (capteurs, tests, imagerie, etc.). Contrairement aux solutions qui demandent une fusion de flux orchestrée manuellement, le **DSL** laisse les pondérations  $\omega_{i,j}$  évoluer en fonction de la **synergie** locale, formant des sous-groupes pertinents sans pipeline imposé.

Cette approche confère une **flexibilité** face aux données partielles ou intermittentes. Des capteurs sujets à des interruptions ou à des pannes peuvent se retirer du **cluster** principal dès lors que leur **synergie** avec les autres entités s'effondre, puis se réinsérer lorsque leur fiabilité s'améliore. On évite ainsi qu'un capteur défaillant ne paralyse l'ensemble.

La **détection de situations inédites** (anomalies, configurations nouvelles) bénéficie particulièrement de la logique non supervisée du **DSL**. Quand un groupe d'entités dérive au regard de la norme, il se regroupe spontanément en **cluster** d'anomalie, sans que l'on ait à pré-définir les classes d'incidents. Cela autorise un repérage de schémas insoupçonnés ou de pannes rares.

Les mécanismes d'**adaptation continue** déjà décrits (section 1.5.4) favorisent la réactivité en contexte dynamique : les pondérations  $\omega$  se réajustent si l'état du patient se modifie (en milieu médical) ou si la chaîne de production passe à un régime nouveau (en milieu industriel). Ce fonctionnement n'exige pas un **réapprentissage** exhaustif à chaque changement.

Cependant, le **principal défi** réside dans la **taille** du graphe et la croissance exponentielle du nombre de liens  $\{\omega_{i,j}\}$  quand les entités se multiplient (nombre élevé de patients, de capteurs, de variables, etc.). Il convient d'appliquer des mesures de **parsimonie** (un seuil en-deçà duquel on supprime des connexions), des procédures de **sparsification** dynamique, ou encore des **misés à jour** localisées (échantillonnage partiel, focus sur un sous-ensemble d'entités), pour maintenir la **scalabilité** et la réactivité du système.



## Conclusion

En **surveillance**, **diagnostic médical** ou **détection d'anomalies**, le **Deep Synergy Learning** apporte une **vision dynamique, distribuée, et auto-organisée** pour :

- Détecter des patterns normaux (clusters stables) vs. anormaux (entités isolées ou clusters dérivants),
- Gérer de multiples sources de données (capteurs, tests, imagerie) sans pipeline prédéfini,
- Réagir en continu aux évolutions ou aux nouvelles situations,
- Combiner éventuellement un **signal** de récompense ou de **validations** cliniques/industrielles partielles pour renforcer l'auto-organisation.

Ainsi, le DSL offre un **mécanisme** mathématique de **pondérations adaptatives** et de **synergies** exploitables dans divers secteurs où la **variabilité** des conditions et la **nécessité** de détecter des comportements hors normes sont prépondérantes.

### 1.6.6. Planification et Optimisation dans l'Industrie 4.0

Les secteurs industriels évoluent vers une **connectivité** accrue, où les machines, capteurs, robots, stocks et systèmes de gestion communiquent en temps réel : c'est la vision de l'**Industrie 4.0**. Les défis qui en découlent — **planification** distribuée, **optimisation** des flux, **maintenance prédictive** et **gestion adaptative** — peuvent se révéler extrêmement complexes. Les méthodes traditionnelles (ordonnancement déterministe, programmation linéaire classique, heuristiques fixes, etc.) montrent souvent leurs limites face à la **dynamique** croissante (marchés volatils, nouvelles commandes urgentes, pannes inattendues). Dans ce cadre, le **Deep Synergy Learning (DSL)**, avec son **mécanisme d'auto-organisation** et de **synergie adaptative**, ouvre de nouvelles pistes pour réaliser une **planification plus flexible** et des **schémas d'optimisation** capables de se réajuster en continu.

Dans cette section (1.6.6), nous examinons :

25. Les raisons pour lesquelles l'**Industrie 4.0** exige un modèle **évolutif** et **distribué**,
26. Comment le **DSL** modélise des entités (machines, stocks, commandes, flux de transport) reliées par des **pondérations synergiques**,
27. De quelle manière ces liens évoluent pour **planifier** et **optimiser** les ressources en temps réel,
28. Des exemples concrets où le **DSL** peut surpasser un pipeline d'optimisation statique, grâce à sa **dynamique d'adaptation** et à sa capacité de **clustering** auto-organisé.

#### 1.6.6.1. Les Défis Industriels dans un Environnement 4.0

Dans l'Industrie 4.0, on fait face à une **abondance** de composants interdépendants. Les **machines** possèdent chacune un état (vitesse, disponibilité, pannes potentielles), tandis que les **chaînes de production** sont subdivisées (stations, lignes, buffers, flux de pièces). Les **stocks** (pièces détachées, semi-finis, produits finis) changent au fil des demandes, et le **transport** peut impliquer différents véhicules (AGV, drones, camions). Enfin, les **ordres de fabrication** évoluent dès lors que des clients passent de nouvelles commandes ou modifient leur planning.

Dans les approches classiques d'**ordonnancement** ou de **planification** (flow shop, job shop, heuristiques variées), on définit un **problème combinatoire** qu'on tente de résoudre sous des **hypothèses** de stabilité. En pratique, l'environnement est loin d'être statique : les machines peuvent tomber en **panne**, un retard logistique perturbe la disponibilité d'une pièce critique, de nouvelles **commandes** arrivent à l'improviste et la **capacité** machine peut varier pour cause de maintenance ou de changement d'outillage. Chaque perturbation requiert souvent un **recalcul** important, voire une réinitialisation d'un algorithme d'ordonnancement.

Le **Deep Synergy Learning (DSL)** propose un **changement de perspective**. Au lieu de modéliser un ordonnancement global, on conçoit chaque **composant** — machine, ordre, stock, mode de transport — comme une **entité**  $\mathcal{E}_i$ . Les **liaisons**  $\omega_{i,j}(t)$  traduisent la **coopération** ou la **complémentarité** entre entités. Par exemple, deux machines  $\mathcal{E}_{\text{machineA}}$  et  $\mathcal{E}_{\text{machineB}}$  voient leur lien s'intensifier si elles se passent fréquemment des pièces sans blocage, ou si l'on constate un gain productif à enchaîner les opérations A puis B. Un **stock**  $\mathcal{E}_{\text{stock}}$  et un **transport**  $\mathcal{E}_{\text{AGV}}$  coopèrent s'ils se synchronisent efficacement pour déplacer des lots. De même, un **ordre**  $\mathcal{E}_{\text{ordre}}$  accroît sa synergie avec une machine spécifique si cette dernière exécute la tâche en temps et qualité optimales.

Lorsqu'un **incident** surgit (panne machine, retard de livraison, surcharge de commande), la **dynamique d'auto-organisation** du DSL réévalue les liaisons  $\omega$ . Un composant en panne voit ses liens se dégrader, réduisant son impact dans le flux. Inversement, si une nouvelle machine arrive ou si un transport reprend sa disponibilité, on insère une **entité** inédite dans le **SCN**, laissant la synergie se construire là où un **gain** apparaît. Cette **reconfiguration** s'effectue **localement**, évitant de relancer un algorithme d'ordonnancement global. Le **réseau** s'ajuste de manière "vivante", reflétant la réalité mouvante de l'Industrie 4.0.

#### 1.6.6.2. Représentation des Flux et Clusters Auto-Organisés

Dans une optique d'**Industrie 4.0**, la mesure de **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  entre deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  peut prendre plusieurs formes. Les machines, les stocks ou les ordres de production entretiennent par exemple un **taux de transfert** élevé lorsque la première fournit régulièrement des pièces à la seconde sans créer de file d'attente. Un **ordre** de production se révèle quant à lui **complémentaire** d'un stock donné si cet ordre requiert précisément les ressources contenues dans ce stock. Par ailleurs, la **performance** d'une association machine–ordre s'accroît lorsque le couple aboutit à un temps de cycle réduit et à un rendement supérieur. Lorsque la **synergie** apparaît remarquable entre deux entités, la pondération reliant ces entités se trouve renforcée (voir section 1.4.5 sur la mise à jour locale des pondérations).

À mesure que certaines combinaisons s'avèrent profitables (machine–stock–ordre–transport, par exemple), leur coopération se **solidifie**, faisant émerger des **clusters** ou sous-réseaux. Chaque **cluster** correspond alors à une configuration de production considérée stable ou efficace, puisqu'il regroupe des entités dont les pondérations mutuelles  $\omega$  sont élevées. On peut ainsi observer la cristallisation d'un groupe {Machine1, StockA, TransportX, OrdreB} au motif que ces éléments forment un **flux** fluide : la machine 1 exploite les ressources du Stock A, le Transport X assure rapidement la logistique, et l'Ordre B en tire un bénéfice opérationnel.

Dans le cadre d'une planification traditionnelle, on conçoit des **lignes** ou **cellules** de production de manière analytique. Le **Deep Synergy Learning (DSL)**, lui, opère un mécanisme **auto-organisé** : les liaisons se **renforcent** quand un groupe de machines s'avère compatible ou lorsqu'un module de transport complète efficacement la chaîne, ce qui fait naître un **cluster** de manière spontanée. Quand un composant (machine, stock ou transport) **dysfonctionne** ou s'avère insuffisamment utile, la **synergie** avec le reste du réseau chute et le composant se trouve mis à l'écart. De nouveaux liens peuvent alors se former vers des entités de **remplacement**, réorientant la production de façon plus souple qu'un pipeline rigide. Ce phénomène procure à l'ensemble une **plasticité** qui évoque un système "**vivant**", réactif aux perturbations et aux changements d'outillage ou de répartition des tâches, conformément à l'esprit de l'Industrie 4.0 décrite en section 1.6.6.1.

#### 1.6.6.3. Approche Dynamique : Adaptation en Temps Réel

Dans un environnement **Industrie 4.0**, le système doit composer avec un **flux** permanent d'événements : nouvelles commandes (urgentes ou non), alertes de stocks bas, maintenance planifiée sur une machine, retards logistiques. Les algorithmes de planification classiques (règle de Johnson, heuristiques de scheduling, etc.) se montrent performants pour un **problème** donné (flow shop, job shop), mais perdent en **flexibilité** lorsqu'on modifie le problème à la volée, car il faut alors réinitialiser ou réactualiser le schéma de planification.

Au **Deep Synergy Learning (DSL)**, la **dynamique d'auto-organisation** se renouvelle en continu. Les **pondérations**  $\omega_{i,j}(t)$  reliant des entités (machines, ordres, stocks, transports...) se remanient itération après itération en fonction de la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$ . Lorsqu'un **ordre urgent** survient, le système introduit une entité  $\mathcal{E}_{\text{ordreUrgent}}$  dans le **Synergistic Connection Network**, et celle-ci voit ses **liaisons**  $\omega_{\text{ordreUrgent},k}$  se renforcer rapidement dès que l'on identifie un gain potentiel avec des machines ou des transporteurs capables de le traiter à temps. Si, au contraire, un transport  $\mathcal{E}_{\text{transport}}$  se montre sous-dimensionné (accumulation de pièces, délais excessifs), sa **synergie** avec le reste du réseau chute, ce qui oriente la production ou la logistique vers d'autres entités plus aptes à satisfaire la demande.

Le **DSL** ne vise pas à figer une **solution** optimale à un instant  $t$ , comme le ferait un algorithme de scheduling : il **maintient** plutôt un **réseau** dont la topologie (clusters de machines, ordres, stocks) évolue au fil des événements. Si l'on dispose d'un **signal** global de performance (par exemple, un throughput, un temps de cycle moyen, un taux de satisfaction client), on peut l'incorporer pour ajuster la **synergie** : les entités participant à une configuration jugée efficace voient leurs liaisons  $\omega$  se consolider, celles menant à un échec ou à une moindre performance s'affaiblissent. Le système s'adapte donc en continu, s'approchant d'une forme d'**auto-organisation** où chaque entité veille à coopérer localement, sans reposer sur une refonte complète de la planification à chaque nouvel incident.

#### 1.6.6.4. Cas d'Exemple : Usine Modulaire

Dans une **usine** où les postes sont **reconfigurables** et la logistique assurée par des **AGV** (véhicules autonomes), les produits fabriqués peuvent changer selon la **demande**. Le **Deep Synergy Learning (DSL)** s'avère alors bénéfique pour une **gestion** plus flexible, en remplaçant ou en complétant les méthodes de planification traditionnelles. On peut définir un **réseau** d'entités  $\{\mathcal{E}_{\text{Poste1}}, \mathcal{E}_{\text{Poste2}}, \dots, \mathcal{E}_{\text{AGV1}}, \mathcal{E}_{\text{AGV2}}, \dots, \mathcal{E}_{\text{Stock1}}, \dots\}$  et des entités "ProduitA", "ProduitB" (avec la possibilité d'en ajouter de nouveaux). Chaque poste ou AGV entretient des **liaisons**  $\omega_{i,j}$  qui se **renforcent** ou se **délient** selon la **synergie** constatée. Si un **poste** particulier traite un certain **produit** avec un taux de réussite élevé, la pondération  $\omega_{\text{poste}, \text{Produit}}$  augmente. De même, un **AGV** se connecte aux postes ou stocks avec lesquels il collabore efficacement, formant ainsi un **cluster** {AGV1, Poste1, Poste2, Stock} si ce groupe de composants assure un flux fluide de pièces.

Lorsqu'on introduit un **nouveau** produit (par exemple "ProduitC"), la synergie se construit **naturellement** autour des postes adéquats. Le **SCN** évolue sans qu'il soit indispensable de reconfigurer entièrement le pipeline ou de lancer un **solveur** d'ordonnancement intégral. Les entités reliées à l'ancien produit peuvent **conserver** leurs liens, tandis que les postes ou AGV inadaptés à "ProduitC" voient leur pondération chuter et se déconnectent graduellement du nouveau cluster. Cette **dynamique** assure une **réactivité** notable : si une **panne** survient sur Poste1, la pondération  $\omega_{AGV, Poste1}$  décroît et l'AGV peut se rediriger vers un autre poste ( $\mathcal{E}_{Poste3}$ ), dont la synergie montera en prenant le relais.

Ce **mécanisme d'auto-organisation** continue favorise la **tolérance** aux pannes et l'adaptation à l'introduction de nouvelles gammes de produits. Les entités inopérantes se "détachent" naturellement (perdent leurs liaisons), et les nouvelles entités intéressantes (nouveaux postes, nouveaux produits) trouvent leur place dans le **Synergistic Connection Network** au fil de la **dynamique** des pondérations. Cela évite la nécessité de **recalculer** périodiquement un ordonnancement global ou de procéder à des heuristiques lourdes, tout en exploitant les **principes** du DSL (sections 1.4.3 et 1.5.4) pour maintenir la cohérence globale du système.

#### 1.6.6.5. Limites et Évolutions : Complexité, Parcimonie, Hybridation

Dans un **environnement industriel** où l'on introduit sans cesse de nouvelles entités (machines, stocks, ordres...), la **taille** du graphe  $\{\omega_{i,j}\}$  peut rapidement devenir prohibitive. Pour préserver la faisabilité, il est essentiel d'envisager des stratégies de **parcimonie** (seuils, coupe des liens trop faibles) ou des **algorithmes locaux** où seules les pondérations pertinentes se mettent à jour, évitant ainsi un recalcul exhaustif à chaque itération. Des **contraintes** peuvent également exclure certains liens d'emblée (une machine non qualifiée pour un ordre, ou un transport incapable de prendre en charge un certain poids), imposant une **pénalisation** ou un **masquage** dans la fonction de synergie.

De plus, lorsque des **règles logiques** (contraintes de sécurité, de qualité) s'ajoutent, on peut intégrer des **entités symboliques** (cf. section 1.5.7). Le **DSL** considère alors la **coopération** entre ces entités symboliques et les entités opérationnelles (machines, ordres...), de sorte que la formation d'un **cluster** illégal (par exemple, un groupement violant une norme de sécurité) n'aboutisse jamais, faute de synergie. Cette hybridation **neuro-symbolique** enrichit la dynamique locale ( $\omega$ ) tout en respectant les cadres réglementaires ou métier.

Il est par ailleurs envisageable de **combiner** le **DSL** à des heuristiques ou solveurs d'ordonnancement établis. Une possibilité consiste à laisser l'**auto-organisation** déterminer un sous-ensemble de ressources potentiellement efficaces, puis à faire appel à un **solveur** classique pour finaliser le planning sur ce sous-ensemble, réduisant la dimension du problème. À l'inverse, un solveur peut fournir une solution initiale, que le **DSL** ajuste localement au gré des aléas (pannes, nouvelles commandes). Cette **collaboration** exploite la force de l'**auto-organisation** (réactivité, plasticité) tout en bénéficiant de méthodes d'**optimisation** éprouvées pour parfaire le résultat. Ainsi, la **plasticité** du DSL et la **rigueur** de l'optimisation se conjuguent pour répondre aux exigences variées de l'Industrie 4.0.

#### Conclusion

Dans le contexte de **planification** et d'**optimisation** pour l'**Industrie 4.0**, le **Deep Synergy Learning** introduit :

- Une **approche distribuée** : chaque ressource (machine, transport, stock) agit comme une entité évolutive, plutôt qu'un simple paramètre dans un solveur central,
- Une **mise à jour adaptative** : plus besoin de résoudre un problème complet à chaque événement, on ajuste localement les synergies pour redistribuer les flux,
- Une **auto-organisation** de clusters (sous-réseaux de production) susceptibles de se reformer si des pannes ou des variations de demandes surviennent,
- Une **tolérance** aux données manquantes ou bruitées (capteur incertain, etc.),
- Une **extensibilité** : on peut ajouter de nouvelles machines ou de nouvelles gammes de produits, et laisser le DSL découvrir où les insérer via la synergie.

Bien sûr, des défis (taille du problème, besoin de parcimonie, contraintes symboliques) subsistent, mais la **philosophie** du DSL — laisser la dynamique  $\omega$  déterminer en continu les regroupements et la répartition des tâches — se marie avec l'exigence de **flexibilité** et d'**évolution** rapide propres à l'Industrie 4.0. Ainsi, le DSL peut constituer un **cadre** pour des solutions d'**ordonnancement** et d'**optimisation** plus "organiquement adaptatives" que les méthodes statiques d'hier, ouvrant la voie à une production réellement **agile** et **intelligente**.



### 1.6.7. Perspectives pour la Recherche Fondamentale en IA Forte

Les précédentes sous-sections (1.6.1 à 1.6.6) ont démontré la **portée pratique** du **Deep Synergy Learning (DSL)** dans divers domaines applicatifs (vision, audio, robotique, recommandation, diagnostic, etc.). Cependant, l'**ambition** du DSL ne se limite pas à une simple amélioration des performances ou de la flexibilité dans ces tâches spécialisées. À un niveau plus **fondamental**, l'approche synergique **ouvre des pistes** de recherche susceptibles de rapprocher l'IA de la notion d'**IA Forte** (ou **IA Généralisée**), c'est-à-dire d'une intelligence capable d'**apprendre** et de **raisonner** de manière autonome et générale, au-delà de cadres strictement définis. Cette section (1.6.7) se propose d'explorer ces **perspectives** théoriques et conceptuelles :

29. L'idée que le DSL, grâce à ses **mécanismes d'auto-organisation**, évoque un fonctionnement plus proche de **systèmes cognitifs** (cerveau, écologies d'informations),
30. La possibilité d'**intégrer** simultanément des dimensions sub-symboliques (apprentissage sur données massives) et des **aspects symboliques** ou logiques (voir 1.5.7),
31. Les **propriétés émergentes** (représentations, micro-réseaux cognitifs, auto-adaptation) qui dépassent la simple exécution d'une tâche,
32. Les **défis mathématiques** et philosophiques associés à la poursuite d'une IA Forte via le DSL.

#### 1.6.7.1. Au-delà de l'Apprentissage Supervisé : Vers l'Auto-Construction de la Connaissance

Une large part des avancées récentes en **IA** se fonde sur l'**optimisation globale** de réseaux neuronaux massifs par **rétropropagation**, qu'il s'agisse de données annotées en abondance ou d'un pré-entraînement auto-supervisé (approche de type GPT, BERT). Bien qu'efficaces, ces modèles conservent plusieurs caractéristiques :

Ils s'appuient sur une **architecture** linéaire ou faiblement hiérarchisée (même si l'on observe des multi-têtes d'attention). Ils dépendent d'une **descente de gradient** ou d'une procédure équivalente visant à minimiser un coût défini globalement. Ils se révèlent peu **adaptables** localement : lorsque la distribution se modifie, on doit souvent procéder à un réapprentissage (fine-tuning), risquant un écrasement partiel des acquis (catastrophic forgetting).

Le **Deep Synergy Learning (DSL)** présente une **dynamique d'auto-organisation** plus proche de l'esprit de **systèmes complexes** (biologiques ou écologiques), où chaque **entité** (sous-module, neurone, flux d'information) gère ses **liaisons** suivant la **synergie** qu'il perçoit avec d'autres entités. Plutôt que de minimiser une **unique fonction de coût**, on autorise l'émergence de **clusters** et la **reconfiguration permanente** des connexions  $\omega_{i,j}$ . Cette approche "**bottom-up**" s'avère potentiellement féconde pour développer des **mécanismes cognitifs** plus avancés et moins tributaires d'un lourd apprentissage supervisé.

Dans une perspective d'**IA Forte**, on envisage souvent un agent qui **accumule** graduellement des connaissances sans les oublier, tout en **réinterprétant** ces savoirs au fil des expériences. Le **DSL** y répond en laissant chaque **entité** conserver sa mémoire locale (paramètres, historique) et en **réévaluant** les liaisons  $\omega$  de façon continue. On obtient ainsi un processus **incrémental** :

De nouvelles **entités** (concepts, données) apparaissent ;

Des **liaisons** se **créent** ou se rompent suivant la pertinence ou la coopération mesurée ;

Des **clusters** de connaissances se **stabilisent**, ou se scindent si leur synergie interne décline ;

L'agent peut alors "**auto-structurer**" ses connaissances, analogue à un cerveau formant et dissolvant des **assemblées** neuronales.

Cette organisation, plus **distribuée** et **évolutive**, pose les bases d'une **auto-construction** de la connaissance, où l'on n'est plus tributaire d'un unique label ou d'une unique fonction de perte, mais d'une dynamique **locale** entre les **entités** qui renforcent ou amoindrissent leurs liens au gré de leur coopérativité. Elle suggère la possibilité d'un **savoir** plus flexible et cumulatif, réduisant la dépendance à un entraînement exhaustif et favorisant une **adaptation** continue aux données nouvelles ou aux changements de contexte.

#### 1.6.7.2. Vers une Cognition Distribuée et Émergente

Le **Deep Synergy Learning (DSL)** trouve ses racines (section 1.3.1) dans l'observation des **systèmes biologiques** exhibant de la **plasticité** (synaptique chez les neurones), ou des phénomènes d'**auto-organisation** (colonies d'insectes, tissus cellulaires). Dans la quête d'une **IA Forte**, on peut s'appuyer sur un **réseau** d'entités en tant que **substrat cognitif**, où :

Les **entités**  $\mathcal{E}_i$  incarnent des briques d'information variées : qu'il s'agisse de **features** perceptuelles (captées par des flux sensoriels), de **concepts abstraits** (catégories, notions sémantiques) ou même de **modules symboliques** (ensembles de règles logiques, sous-réseaux dédiés).

Les **liaisons synergiques**  $\omega_{i,j}$  instaurent un **renforcement** de type “hebbien généralisé” dès lors qu’une **coopération** entre entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  procure un **gain** (qu’il s’agisse de performance, de co-information ou de simplification).

Les **clusters** qui émergent (assemblées ou circuits cognitifs) assument des rôles spécifiques dans le traitement conceptuel, la mémoire de travail, la planification ou encore l’intégration de signaux multiples.

Au sein de ce **substrat**, chaque **cluster** reflète potentiellement une micro-théorie ou un schéma conceptuel, regroupant des entités pour former une **hypothèse**, un **concept**, voire un **plan d’action**. La structuration se fait de façon **dynamique** : les assemblées apparaissent ou se scindent selon l’évolution de leur synergie, permettant de reconfigurer en continu l’inventaire des connaissances ou des idées. Contrairement aux architectures statiques (couches fixes), cette configuration s’**auto-régule**, entre **plasticité** (création et rupture de liens) et **stabilité** (clusters pertinents).

En focalisant l’analyse sur la **dynamique** globale, on peut reproduire certains **processus** cognitifs : une **forme d’attention** résulte si un sous-réseau monopolise la synergie disponible, comparable à un focus transitoire. Des **boucles** de rétroaction positives ou négatives peuvent générer des **oscillations**, analogues à des rythmes cognitifs ou des alternances d’états mentaux. La **conservation** d’équilibres permet au système de préserver des organisations robustes, tout en se montrant apte à accueillir de nouvelles liaisons ou à rompre les moins pertinentes.

De tels phénomènes, absents d’un CNN ou d’un Transformer (qui restent malgré tout dans une structure de pipeline, bien que complexe), s’apparentent plutôt à un **processus** cognitif **distribué**, dans lequel l’information chemine entre assemblées modulaires qui se constituent et se dissolvent librement. On entrevoit là un **jalon** vers des agents en mesure de **construire** et de **réviser** en continu leurs hypothèses ou “théories”, de manière plus proche de la **biologie** et des **neurosciences** que de la minimisation stricte d’une fonction de coût globale.

### 1.6.7.3. Intégration Symbolique : Socle pour le Raisonnement Abstrait

Dans une démarche d’**IA Forte**, il ne suffit pas de traiter des signaux sub-symboliques (capteurs, images, séquences, etc.) : il faut également pouvoir manier des **représentations abstraites** et des règles logiques complexes. Les réseaux neuronaux profonds, malgré leur puissance pour l’apprentissage sur données massives, peinent souvent à exécuter un raisonnement explicitement symbolique, à moins d’y adjoindre des modules spécialisés. Le **Deep Synergy Learning (DSL)** propose un **cadre unifié** où des entités clairement “symboliques” (comme  $\mathcal{E}_{\text{rule}}$ ,  $\mathcal{E}_{\text{concept}}$ ) coexistent avec des entités sub-symboliques ( $\mathcal{E}_{\text{sensor}}$ ,  $\mathcal{E}_{\text{feature}}$ ), comme décrit à la section 1.5.7. Les liaisons  $\omega_{\text{rule},\text{sensor}}$  s’enrichissent dès lors qu’une **règle** s’applique à un **schéma** concret observé : la synergie se renforce si la correspondance entre la règle et la configuration sensorielle se révèle fructueuse (information mutuelle, gain de performance, etc.). On obtient alors un **micro-réseau** dans lequel un ensemble de règles se coordonne avec un ensemble d’observations, sans cloison rigide séparant le module logique du module perceptif.

Dans cette perspective, la **dynamique** des pondérations  $\omega$  ne se contente plus de rapprocher ou de séparer des entités “vision” ou “audio” : elle peut tout autant consolider des liens entre un **concept abstrait** et divers indices sensoriels confirmant sa validité, ou inversement discréditer une règle en la séparant d’un environnement où elle ne s’applique pas. Cela ouvre la voie à un **raisonnement** logico-perceptif plus fluide : lorsqu’une hypothèse symbolique  $\mathcal{E}_{\text{rule}}$  se voit soutenue par l’observation  $\mathcal{E}_{\text{sensor}}$ , leur liaison grandit et un **cluster** naît, fusionnant la dimension symbolique et la dimension sub-symbolique. Cette approche épargne la nécessité d’implanter un connecteur artificiel entre un réseau neuronal et un moteur logique indépendant : c’est le **réseau** lui-même, via la mise à jour des synergies, qui unifie ces deux registres. On y gagne en transparence et en **plasticité** : les règles symboliques peuvent être admises, ajustées ou rejetées au gré de la compatibilité avec les données sensorielles, se reformant ou se dissolvant au sein du **SCN**. Il en découle un possible **socle** pour un raisonnement abstrait mieux ancré dans la perception, condition importante de l’**IA Forte**, qui réclame un agent sachant à la fois extraire des modèles du monde et manipuler des constructions logiques plus élevées.

### 1.6.7.4. Défis Mathématiques : un Pas vers une Théorie de la Cognition Synergique

Le **Deep Synergy Learning (DSL)** peut se voir comme un **système dynamique** non linéaire de grande dimension, dont la **matrice**  $\Omega(t)$  (ou un hyper-graphe) évolue suivant une loi similaire à

$$\Omega(t+1) = \Omega(t) + \eta [S(\Omega(t)) - \tau \Omega(t)],$$

où  $\Omega(t)$  désigne l’ensemble des pondérations  $\{\omega_{i,j}(t)\}$  à l’instant  $t$ , et  $S(\Omega(t))$  symbolise la mesure de **synergie** calculée entre les entités, qui peut elle-même dépendre de l’état global. Cette dynamique est susceptible de générer des **attracteurs** multiples (plusieurs organisations cognitives possibles), des **bifurcations** lorsque les synergies changent brusquement, voire des **cycles limites** interprétables comme des rythmes cognitifs ou des alternances d’états mentaux. L’**analyse** formelle (existence et stabilité d’attracteurs, transitions critiques, phénomènes d’hystérèse) reste toutefois un **défi** majeur, notamment si l’on souhaite déployer de nombreuses entités ou autoriser une définition non triviale de la synergie (information mutuelle n-aire, PID, etc.).

Pour gérer la croissance potentiellement explosive du nombre de liaisons  $\{\omega_{i,j}\}$  et leur mise à jour en temps réel, on se tourne fréquemment vers des **méthodes de parcimonie** (seuils, coupes, hiérarchies de clusters) ou des **stratégies** d’échantillonnage. Certains

mécanismes inspirés de la **biologie** — par exemple un terme d’oubli rapide ou une saturation — contribuent également à la **scalabilité**. Le choix précis de la fonction de synergie  $S$ , de la règle de mise à jour et de la topologie initiale influe sur la façon dont le réseau s’organise.

En parallèle, les travaux en **sciences cognitives** ou neurosciences computationnelles (comme la **théorie de l’information intégrée** de Tononi) proposent de quantifier la **part** d’information réellement unifiée dans un système. Dans l’esprit du **DSL**, on peut définir une **intégration globale** à partir de la **somme** des synergies (ou d’une co-information n-aire) :

$$\text{Intégration globale} = \sum_{i,j} \omega_{i,j} S(\mathcal{E}_i, \mathcal{E}_j),$$

ou encore effectuer une version plus générale au sein d’un **cluster** n-aire. Cette démarche veut évaluer la **“richesse”** émergente dans un réseau en perpétuelle auto-organisation. Elle ouvre un **pas** vers des réflexions sur la **conscience artificielle** ou l’**autonomie** inspirée de la cognition humaine, puisqu’elle quantifie à quel degré le réseau regroupe et **intègre** l’information sous forme de micro-réseaux cognitifs, susceptibles d’exhiber des propriétés d’**émergence** et de **complexité** caractéristiques des systèmes vivants.

## Conclusion

Les perspectives du **Deep Synergy Learning** pour la **recherche fondamentale** en **IA Forte** s’articulent autour de :

- Sa **capacité** à s’**auto-structurer** hors d’un pipeline rigide ou d’une unique fonction de coût globale,
- Son **paradigme** distribué et émergent, plus proche de certains **modèles biologiques** ou cognitifs,
- L’**intégration** simultanée de dimensions sub-symboliques (features, embeddings) et symboliques (règles, concepts logiques) dans un **même** réseau de synergie,
- L’éventuelle **quantification** de la **richesse** et du **niveau d’intégration** du réseau, ouvrant la voie à des notions plus avancées (auto-réflexion, conscience, adaptativité illimitée).

En somme, si le **DSL** se montre déjà **efficace** et **agile** dans des applications concrètes (sections 1.6.1 à 1.6.6), il pourrait aussi servir de **fondement** à une IA plus **globale**, capable d’apprendre, de se réorganiser et d’**intégrer** différents registres de représentations (perception, logique, symbolique) de manière plus **organique**. La réalisation d’une **IA Forte** n’est pas garantie, mais le **DSL** suggère une nouvelle **philosophie** : remplacer la hiérarchie imposée par une **auto-organisation** continue, potentiellement apte à donner naissance à des **capacités cognitives** plus vastes et plus proches du fonctionnement du cerveau ou de systèmes biologiques.

### 1.7. Défis, Contraintes et Ouvertures

Les sections précédentes (1.1 à 1.6) ont mis en lumière la **logique fondamentale** du **Deep Synergy Learning (DSL)**, ses **mécanismes** d’auto-organisation et ses **capacités** à apporter des solutions nouvelles dans de multiples domaines applicatifs (vision, audio, robotique, recommandation, diagnostic, etc.). Toutefois, comme toute approche novatrice, le DSL n’échappe pas à des **défis** et des **contraintes** majeurs. Sur le plan **computationnel, théorique, méthodologique** et **éthique**, de nombreuses questions demeurent ouvertes, exigeant une recherche soutenue pour perfectionner le paradigme et garantir son insertion fluide dans des environnements industriels, médicaux, ou encore scientifiques.

La volonté du DSL de fonctionner sans architecture hiérarchique figée, et de laisser les entités s’**auto-organiser** via des synergies, requiert notamment :

- Une **scalabilité** adéquate (comment passer à des milliers ou millions d’entités sans explosion de la complexité ?),
- Une **qualité** et une **accessibilité** suffisantes des données (pour estimer la synergie de façon robuste),
- Des **algorithmes d’optimisation** aptes à gérer des systèmes dynamiques, non linéaires et potentiellement hétérogènes,
- Un **contrôle** de la stabilité et de la convergence (éviter les oscillations ou l’effondrement du réseau),
- Une **interprétabilité** pour l’humain, d’autant plus cruciale lorsque les décisions impactent des vies ou des processus critiques,
- Des **considérations éthiques** et réglementaires pour encadrer l’usage de systèmes auto-organisés, notamment en termes de responsabilité, d’acceptabilité, ou de gestion de biais,
- Des **comparaisons expérimentales** rigoureuses avec les approches existantes (réseaux neuronaux profonds, méthodes symboliques, algorithmes d’optimisation classiques) pour valider la pertinence et les conditions d’efficacité du DSL.

Dans cette section (1.7), nous examinons un à un ces **défis, contraintes et ouvertures**, déclinés en sept points :

33. **Complexité Computationnelle et Scalabilité** (1.7.1)
34. **Qualité et Disponibilité des Données** (1.7.2)
35. **Développement d’Algorithmes d’Optimisation Appropriés** (1.7.3)
36. **Contrôle et Stabilité des Processus Auto-Organisés** (1.7.4)
37. **Interprétabilité et Explicabilité pour l’Humain** (1.7.5)
38. **Considérations Éthiques et Réglementaires** (1.7.6)
39. **Comparaisons Expérimentales avec d’Autres Approches** (1.7.7)

Ce panorama permettra de saisir la **complexité** inhérente à l’approche synergique, tout en mettant en relief les **axes** de recherche et les **collaborations** scientifiques qu’elle appelle, pour faire du DSL un cadre de plus en plus solide dans l’écosystème de l’IA et de l’apprentissage automatique.

#### 1.7.1. Complexité Computationnelle et Scalabilité

La première critique ou préoccupation qu’on peut formuler à l’égard du **DSL** concerne la **taille** potentielle du **graphe** des entités et la **complexité** des opérations nécessaires pour **calculer** ou **mettre à jour** les **pondérations** synergiques ( $\omega_{i,j}$ ). En effet, si on postule :

- $n$  entités  $\{\mathcal{E}_1, \dots, \mathcal{E}_n\}$ ,
- une synergie binaire  $S(\mathcal{E}_i, \mathcal{E}_j)$  pour chaque paire  $(i, j)$ ,
- une mise à jour régulière (à chaque itération ou chaque batch temporel),

alors la **complexité brute** peut facilement atteindre  $O(n^2)$  par itération, ce qui devient ingérable quand  $n$  est de l’ordre de plusieurs milliers, voire millions, de composantes. De plus, si l’on veut aller vers des **synergies n-aires** (pour capturer des coopérations à plusieurs entités simultanément), la situation s’aggrave encore (complexité exponentielle).

### 1.7.1.1. Problématique du “Tout Relier à Tout”

L’un des principes centraux du **Deep Synergy Learning (DSL)** est de laisser les **entités**  $\{\mathcal{E}_i\}$  se connecter ou se déconnecter au fil du temps, suivant la **synergie** perçue. Pourtant, permettre qu’une entité puisse se relier à toutes les autres sans limite conduit vite à un **graphe complet**, où chaque paire  $(i, j)$  exige un calcul de synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$ . Cette démarche devient rapidement ingérable :

40. Le calcul de  $S(\mathcal{E}_i, \mathcal{E}_j)$  pour toutes les paires  $(i, j)$  croît en  $O(n^2)$ .
41. La mise à jour des pondérations  $\omega_{i,j}$  (section 1.4.5) présente la même complexité en  $O(n^2)$ .
42. Le stockage de l’ensemble  $\{\omega_{i,j}\}$  requiert une **mémoire** en  $O(n^2)$ .

Dans des applications **massives** (robotique multi-capteurs, systèmes de recommandation avec des millions d’utilisateurs et de contenus, vision distribuée avec d’innombrables descripteurs), la construction d’un tel graphe complet compromet la **scalabilité** du DSL. Il devient crucial de maintenir une certaine **parsimonie** au sein du réseau, de sorte à ne pas créer aveuglément des liaisons  $\omega_{i,j}$  pour chaque paire.

Une **stratégie** commune consiste à **restreindre** le calcul de synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$  aux seules paires dont les entités se révèlent “**assez proches**” ou déjà **corrélées**. On peut, par exemple, refuser la création de liens si la distance entre  $\mathbf{x}_i$  et  $\mathbf{x}_j$  dépasse un certain **seuil**  $\epsilon$ . On peut aussi confiner le calcul à un **voisinage**  $k$ -plus-proches-voisins, limitant la croissance de connections. Par ailleurs, il faut un **mécanisme** pour découvrir graduellement de nouvelles liaisons, via un **échantillonnage** ou une **partition** initiale, sans explorer d’emblée toutes les paires  $(i, j)$ .

En parallèle, on définit des **règles** pour **supprimer** ou **inactiver** automatiquement les liens jugés trop faibles ou non pertinents (auto-suppression sous un seuil  $\omega_{\min}$ ). De cette manière, la densité du graphe reste modérée. On vise alors une complexité proportionnelle au nombre de liaisons “actives”, ce qui peut être bien inférieur à  $n^2$ . Au bout du compte, le **réseau** se rapproche d’un **k-NN** ou d’un  $\epsilon$ -**radius** dynamique, sur lequel on applique les mises à jour (section 1.4.5). Le **DSL** y opère donc ses principes d’auto-organisation et de synergie, tout en évitant l’écueil d’un graphe complet ingérable.

### 1.7.1.2. Calcul de Synergie : Coût des Mesures Informationnelles

Lorsque la **synergie** prend la forme d’une **co-information** ou d’une **information mutuelle** entre entités modélisées comme variables aléatoires, on se heurte à la difficulté du calcul ou de l’estimation de grande dimension. Les estimateurs non paramétriques (k-NN, Kernel density) sont souvent coûteux et la quantité de données exigée peut augmenter de manière exponentielle avec la dimension (curse of dimensionality). Pour rendre ces métriques d’information pratiquement utilisables, on peut :

43. **Limiter** la dimension ou extraire des **features** plus compactes (par un autoencodeur, par une PCA, etc.), afin de réduire le champ d’action des estimateurs d’entropie et de co-information.
44. **Recourir** à des mesures de synergie plus **légères** dans de nombreux cas (par exemple, la distance euclidienne ou la similarité cosinus), quitte à se priver de la richesse de l’information mutuelle stricte.
45. **Employer** des heuristiques d’**approximation** lorsqu’on tient absolument à l’information théorique, comme un **downsampling** des données, une **approximation** paramétrique (Gaussienne, mixture Gaussienne), ou la construction d’estimateurs allégés.

Ces méthodes aident à concilier la **vision** d’un DSL basé sur la co-information et la **réalité** des applications de grande dimension, pour lesquelles un calcul exact d’entropie resterait prohibitif.

### 1.7.1.3. Mise à Jour Itérative ou Partielle

Même si l’on parvient à maintenir une **structure parcimonieuse** (où un sous-ensemble restreint de liaisons  $\omega_{i,j}$  demeure actif), la **mise à jour** de chaque pondération peut néanmoins devenir un **processus lourd** dans un réseau de grande taille. Pour rendre le **Deep Synergy Learning (DSL)** praticable à grande échelle, il est donc nécessaire de recourir à des **méthodes itératives** ou à des **mises à jour partielles**, évitant un recalcul exhaustif à chaque itération.

Une solution envisageable réside dans une **mise à jour locale en parallèle**. Chaque entité  $\mathcal{E}_i$  ne s’occupe que de ses **voisins** actuels, c’est-à-dire les nœuds auxquels elle est reliée par des liaisons  $\omega_{i,j}$  non négligeables. Chacune calcule et met à jour les pondérations concernant ses connexions, en parallèle avec les autres entités. Cette approche sollicite la **communication distribuée** (ou la mutualisation des ressources, GPU ou TPU), tout en restant localisée : on n’explore pas toutes les paires  $(i, j)$  du réseau, mais seulement les liens existants.

Il demeure toutefois essentiel d’introduire un **mécanisme** pour éviter les conflits et saturations éventuelles. Par exemple, si plusieurs entités cherchent à coopérer en excès avec la même cible, on peut concevoir une **inhibition compétitive** assurant que les liaisons se

stabilisent autour d'un certain nombre de voisins. Certaines idées issues des **systèmes multi-agents** (communication locale asynchrone, règles de stabilisation) favorisent la **convergence** du réseau. Ce paradigme s'apparente finalement à une **simulation** de réseaux biologiques ou neuronaux, où les neurones mettent à jour leurs connexions en temps réel de manière distribuée. La **scalabilité** demeure possible tant que la **densité** du graphe demeure limitée et que les calculs de synergie (ou de similarité) ne concernent pas l'ensemble des paires dans un espace de grande dimension.

#### 1.7.1.4. Vers une Mathématique de la Parcimonie et du Grouping Évolutif

Une autre voie de recherche consiste à définir une **énergie** ou un **coût** global  $\mathcal{J}(\Omega)$  pour le réseau  $\Omega$ , englobant non seulement le terme  $-\sum \omega_{i,j} S_{i,j}$  (valorisant les liens qui affichent une forte synergie) mais également un terme de **pénalisation** du nombre de liaisons non nulles. On peut par exemple écrire :

$$\mathcal{J}(\Omega) = - \sum_{i,j} \omega_{i,j} S_{i,j} + \alpha \|\Omega\|_0,$$

où  $\|\Omega\|_0$  compte le nombre de connexions actives (i.e.  $\omega_{i,j} > 0$ ) et  $\alpha > 0$  contrôle l'importance de la parcimonie. Cette formulation **force** le réseau à **rester** relativement épars, tout en **maximisant** la somme de synergies. L'analyse mathématique des **minima locaux** de  $\mathcal{J}(\Omega)$ , ainsi que la description des **trajectoires** de descente (en temps continu ou discret) dans un espace de grande dimension, pose d'importants **défis** : on souhaite caractériser la **stabilité** des clusters émergents et la **croissance** potentielle de la taille du réseau. Cette approche ouvre des perspectives pour un contrôle plus strict de la **complexité**, en assurant qu'une connectivité limitée ne se transforme pas en un graphe complet, préservant ainsi la faisabilité du **DSL** à grande échelle.

### Conclusion

La **complexité computationnelle** et la **scalabilité** constituent sans doute les **premiers** grands défis pour le **Deep Synergy Learning**. Sans mesures de parcimonie et sans heuristiques de mise à jour partielle, le coût en temps et en mémoire peut devenir prohibitif. Les **pistes** de solutions incluent :

- **Sparsité** intégrée dans la création/rupture de liens,
- **Méthodes** d'estimation approximative des synergies dans des espaces de haute dimension,
- **Parallélisme** et algorithmes distribués,
- **Formulations** d'énergie globale favorisant la parcimonie.

Ces recherches sont cruciales pour permettre au DSL de s'étendre à grande échelle (milliers, millions d'entités) sans perdre l'efficacité ou la réactivité nécessaires dans les applications industrielles, médicales, ou cognitives.

#### 1.7.2. Qualité et Disponibilité des Données

Au-delà des défis de **complexité** et de **scalabilité** (1.7.1), le **Deep Synergy Learning (DSL)** doit aussi relever des enjeux majeurs quant à la **qualité** et la **disponibilité** des données. En effet, la logique synergique et auto-organisée repose sur l'évaluation locale de **mesures de synergie** ( $S(\mathcal{E}_i, \mathcal{E}_j)$ ) et sur la **mise à jour** (positive ou négative) de pondérations  $\omega_{i,j}$ . Lorsque les données sont trop **bruitées**, trop **parcellaires** ou manquent de **représentativité**, la convergence vers des clusters pertinents ou la stabilité de l'architecture peut être compromise. Le DSL, tout en étant plus tolérant aux lacunes qu'un réseau neuronal figé, n'en demeure pas moins dépendant d'un certain **niveau** de fiabilité et de diversité des entrées pour exploiter la synergie.

Dans cette sous-section (1.7.2), nous analyserons :

46. Comment la **qualité** et la **couverture** des données impactent le calcul de la synergie,
47. Pourquoi la **disponibilité** (flux continu vs. données rares) peut influencer sur la dynamique du DSL,
48. Les biais possibles et les problèmes d'**incohérence** qui peuvent fausser l'auto-organisation,
49. Les pistes pour **sécuriser** ou **améliorer** la robustesse de l'apprentissage synergique face à des données imparfaites.

### 1.7.2.1. Importance d'une Bonne Couverture et Diversité des Données

Le **Deep Synergy Learning (DSL)** s'appuie sur la fonction de synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$  pour guider la mise à jour des pondérations  $\omega_{i,j}(t)$ . Lorsque ces données  $\mathcal{E}_i$  et  $\mathcal{E}_j$  sont rares, bruitées ou biaisées, l'estimation de leur **distance** ou **similarité** (ou même de leur **co-information**) risque d'être inexacte. Cela peut entraîner un renforcement abusif de liens qui ne recouvrent aucune complémentarité réelle ou, à l'inverse, un affaiblissement injustifié de liaisons potentiellement utiles. Le réseau peut dès lors former des **clusters** peu pertinents, traduire des **biais** préexistants dans les données ou négliger des sous-groupes minoritaires.

Au-delà du simple problème d'**apprentissage** unitaire, le **DSL** met en relief la **couverture** temporelle. Si les conditions (saisons, configurations) varient, un système cherchant à s'adapter en continu aura besoin d'exemples suffisants pour chaque **régime** de données (périodes spécifiques, sous-populations) afin de réévaluer les synergies. Si l'on ne présente jamais de situations "hiver" lors de la constitution initiale du réseau, il se peut que les liaisons nécessaires ne s'instaurent pas, et le DSL échouera à former des **clusters** adaptés dès l'apparition d'un climat hivernal. La même logique vaut pour d'autres domaines : un **flux** d'information continu requiert un nombre d'**itérations** ou de répétitions suffisant pour que chaque nouvelle modalité ou contexte trouve sa place dans le **Synergistic Connection Network**.

La diversité et la **représentativité** des données apparaissent donc cruciales pour qu'un **DSL** puisse véritablement tirer parti de son **auto-organisation**. Sans cette diversité, il risque de manifester des **biais** similaires à ceux de nombre de méthodes d'apprentissage, la différence étant qu'ici c'est la **formation** des clusters (et la dynamique des liaisons  $\omega$ ) qui pâtit des lacunes dans la couverture des phénomènes et des populations.

### 1.7.2.2. Bruit, Lacunes et Données Incomplètes

Comme discuté en section 1.5.3, le **Deep Synergy Learning (DSL)** fait preuve d'une **flexibilité** remarquable vis-à-vis des données incomplètes ou bruitées. Une entité trop incertaine finit par se **retrouver isolée** si sa synergie avec les autres entités demeure inexistante, préservant ainsi la **structure** principale du réseau. Toutefois, lorsque la majeure partie des entités se révèle **entachée** d'un bruit élevé ou d'un manque important de mesures, l'**auto-organisation** échoue à faire ressortir des synergies réelles : la plupart des pondérations  $\omega_{i,j}$  restent faibles et aucun **cluster** solide ne se forme, ou bien les liens sont au contraire **renforcés** sur la base de signaux erronés (artefacts de bruit).

Le **DSL** reste donc partiellement tributaire d'un **nettoyage** ou d'une **consolidation** préliminaire si le **bruit** se montre trop massif, sous peine de limiter sa capacité à détecter les vraies complémentarités. Plusieurs stratégies peuvent alors intervenir pour filtrer ou rehausser la qualité des données :

- Un **filtrage statistique** peut écarter ou sous-pondérer les entités dont les mesures apparaissent trop incertaines (taux de données manquantes élevé, incohérences récurrentes).
- Une **fusion** locale d'entités similaires peut consolider leur information : si deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$  se recouvrent largement ou présentent des mesures redondantes, il est envisageable de les grouper en une entité plus fiable et plus "complète".
- L'inclusion de **scores** de fiabilité dans la fonction de synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$  évite de s'appuyer lourdement sur des estimations basées sur un faible nombre de points ou sur des valeurs soupçonnées d'être bruitées. Par exemple, on peut moduler la similarité ou la distance d'après un **coefficient de confiance** attaché à chaque entité ou à chaque mesure.

Grâce à ces mécanismes, on endigue l'effet d'un bruit omniprésent tout en préservant la logique d'**auto-organisation** du DSL. Ainsi, même si certaines entités restent partiellement incomplètes, elles peuvent trouver leur place dans le **Synergistic Connection Network** dès lors qu'elles entretiennent au moins quelques synergies robustes avec d'autres entités.

### 1.7.2.3. Accès et Disponibilité Continue : le DSL comme Système en Ligne

Dans de nombreux environnements, il est nécessaire de gérer un **flux** ininterrompu de données et de réajuster en permanence les structures d'apprentissage. Le **Deep Synergy Learning (DSL)**, déjà décrit comme un processus de mise à jour itérative (voir la section 1.4.5), peut fonctionner en **mode "online"** si l'on s'assure que chaque entité  $\mathcal{E}_i$  reçoit ses observations de manière régulière ou semi-régulière. Lorsque de nouvelles mesures sont disponibles, la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  est recalculée localement (ou approximée sur une fenêtre glissante), puis la pondération  $\omega_{i,j}(t+1)$  s'actualise selon l'équation :

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

Cette logique d'évolution incrémentale requiert un **accès** continu aux flux entrants, que l'on traite idéalement par un système de gestion de messages (type broker Kafka) ou par un stockage minimal assurant la conservation des  $\omega_{i,j}(t)$ . Le réseau se reconfigure au fil de la réception de nouvelles données, sans qu'il soit nécessaire de reprendre un apprentissage exhaustif en mode batch. Dans un contexte industriel où la disponibilité des données peut être irrégulière, il arrive parfois que certaines entités n'aient aucune mise à jour pendant



une période. Il faut alors conserver leurs liaisons  $\omega_{i,j}(t)$  inchangées et ne pas pénaliser excessivement l'absence de mesures, conformément aux principes d'adaptation face aux lacunes de données (section 1.5.3). Lorsque la distribution évolue de manière brusque, le **DSL** réévalue les synergies de façon locale et progressive : dès qu'une entité se montre moins corrélée à son voisinage, ses pondérations décroissent et elle se retrouve partiellement isolée, tandis que de nouveaux liens peuvent se former avec d'autres entités plus en adéquation avec la situation nouvelle.

Du point de vue du **déploiement**, un tel système en ligne exige une **synchronisation** ou un **parallélisme** adapté. Chaque entité ne traite que ses connexions actives, ce qui maintient la complexité à un niveau proportionnel au nombre de liaisons conservées (voir la section 1.7.1.2 sur la parcimonie). Pour des applications massivement distribuées, il est possible de faire coexister plusieurs nœuds DSL partiels et de synchroniser leurs matrices  $\omega$  par échanges de sous-graphes ou de résumés statistiques. Cette configuration favorise la **scalabilité** et la **résilience**, car si un nœud local subit une interruption ou un ralentissement, le reste du réseau n'est pas forcé de s'arrêter, et les autres entités continuent à ajuster leurs pondérations. Une telle conception rend le **DSL** apte à gérer des environnements en perpétuel changement, tout en intégrant les principes d'auto-organisation et de mise à jour continue.

#### 1.7.2.4. Biais et Incohérences Structurelles

Dans le **Deep Synergy Learning (DSL)**, les liaisons  $\omega_{i,j}$  émergent localement à travers la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$ . Lorsque les données d'entrée sont biaisées, par exemple avec une surreprésentation de certains profils (sections 1.5.3 et 1.7.2.1), le réseau a tendance à **renforcer** les clusters correspondant aux **groupes majoritaires**, tout en marginalisant des entités issues de groupes minoritaires ou insuffisamment renseignés. Dans des domaines critiques comme le **diagnostic médical** ou la **recommandation sensible**, un tel phénomène peut avoir pour effet d'**entériner** des inégalités ou de mener à des **erreurs** d'interprétation (sous-diagnostic pour certains patients, ou éviction de contenus pertinents pour certaines catégories d'utilisateurs).

À la différence d'un **modèle supervisé** où l'on surveille une **loss** globale sur l'ensemble des données, le DSL base ses ajustements sur des **mesures locales** de synergie. Même si la structure globale  $\Omega$  peut finir par présenter un arrangement biaisé, la démarche d'**auto-organisation** ne possède pas toujours de mécanismes internes de correction. Les entités minoritaires voient leurs pondérations  $\omega_{i,j}$  rester faibles, du fait qu'elles ne rencontrent pas suffisamment de synergie avec la majorité. Par ailleurs, si une **incohérence** surgit ou si un artefact statistique biaisé devient prépondérant, le DSL peut stabiliser un **cluster** aberrant, en l'absence de contrôle externe.

Pour pallier cette difficulté, il est indispensable de prévoir des **mécanismes de validation** ou de **monitoring** global. On peut, par exemple, injecter des **entités symboliques** (sections 1.5.7 et 1.7.2.3) qui portent des règles ou des normes éthiques, limitant la croissance de clusters injustes ou repérant les configurations contradictoires. On peut également imposer un **feedback** externe, ou un signal de performance englobant des critères d'équité, veillant ainsi à empêcher la fermeture de groupes déviants. La structure  $\omega$  peut alors être réévaluée à la lumière de ces **contraintes** d'équité ou de cohérence, conférant au réseau un degré de régulation qui contrebalance les dérives potentielles issues de la simple coopération locale.

#### 1.7.2.5. Conclusion

La **qualité** et la **disponibilité** des données sont cruciales pour le **Deep Synergy Learning**, tout comme pour la plupart des approches d'IA. Néanmoins :

- Le **DSL** peut mieux tolérer la **partialité** (en isolant les entités trop bruitées),
- Il s'adapte mieux à un **flux continu** (mise à jour locale des pondérations),
- Il demeure sujet à des **biais** et à un besoin de **représentativité** dans les données.

Ainsi, un important **travail préparatoire** est souvent nécessaire pour :

50. **Filtrer** ou **pondérer** la qualité des données,
51. Introduire des **scores de confiance** dans la synergie,
52. Établir une **stratégie** de streaming ou de batch mixte pour nourrir la dynamique  $\omega$ ,
53. Surveiller la **formation** de clusters anormaux ou la propagation de biais dans l'auto-organisation.

Réussir ce pilotage des données est fondamental pour tirer profit du DSL : un système auto-organisé ne saurait corriger des écueils majeurs de collecte ou de représentativité sans un minimum d'**ingénierie** et de **gouvernance** des données.



### 1.7.3. Développement d'Algorithmes d'Optimisation Appropriés

Une fois admise la nécessité de gérer la **complexité** (1.7.1) et de soigner la **qualité** des données (1.7.2), se pose la question cruciale des **algorithmes** qui permettront de **piloter** la dynamique du **Deep Synergy Learning (DSL)**. En effet, ce paradigme repose sur l'**auto-organisation** de multiples entités  $\{\mathcal{E}_i\}$  et sur la **mise à jour** de l'ensemble des pondérations synergiques  $\omega_{i,j}(t)$ . Dans le modèle théorique simple, on écrit souvent :

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

Où  $S(\mathcal{E}_i, \mathcal{E}_j)$  désigne la **synergie** entre ces entités.

En pratique, mettre en œuvre cette loi pour un grand nombre d'entités, avec des synergies potentiellement complexes (non linéaires, conditionnelles, n-aires), n'est pas trivial. On a besoin de **méthodes** d'optimisation aptes :

54. À **stabiliser** le système (éviter les oscillations permanentes ou l'explosion des pondérations),
55. À **accélérer** la convergence vers des configurations pertinentes (clusters, macro-clusters),
56. À prendre en compte des **contraintes** (symboliques, de cohérence, de parcimonie),
57. À **s'adapter** quand de nouvelles entités ou données apparaissent (flux continu).

Cette sous-section (1.7.3) discute quelques **pistes** mathématiques et algorithmiques pour rendre le DSL plus solide et plus efficace :

- Des méthodes **locales** (mise à jour distribuée),
- Des formulations **globales** (fonction d'énergie, descente de gradient généralisée, approches de type recuit simulé),
- Des **heuristiques** ou des règles inspirées de la biologie et des systèmes complexes (sélection, reproduction, extinction),
- Des **extensions** pour la synergie n-aire ou conditionnelle.

#### 1.7.3.1. Approche Locale et Distribuée

Dans l'esprit **biologique** et de nombreux modèles de **systèmes complexes**, on peut envisager la mise à jour de chaque pondération  $\omega_{i,j}$  de manière **locale**, sans qu'un contrôle global impose la synchronisation. Chaque entité  $\mathcal{E}_i$  communique uniquement avec ses **voisins** au sens du graphe, autrement dit les entités  $\mathcal{E}_j$  pour lesquelles  $\omega_{i,j}(t)$  est significatif. L'entité  $\mathcal{E}_i$  calcule alors la **synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  sur la base de mesures ponctuelles (ou d'un historique local) et applique la règle :

$$\omega_{i,j}(t+1) = \omega_{i,j}(t) + \eta_{i,j}(t) [S_{i,j}(t) - \tau_{i,j}(t) \omega_{i,j}(t)].$$

Cette équation reprend le formalisme général du **Deep Synergy Learning** (section 1.4.5) en autorisant des coefficients  $\eta_{i,j}(t)$  et  $\tau_{i,j}(t)$  qui dépendent de la paire  $(i, j)$  ou varient dans le temps, procurant une **régulation** fine. L'**avantage** est de rendre l'algorithme **parallélisable**, puisque chaque entité  $\mathcal{E}_i$  peut mettre à jour ses liaisons sans solliciter un "master" central. Une difficulté tient toutefois à la **synchronisation** : si plusieurs entités se mettent à renforcer ou à affaiblir simultanément des liaisons communes, des boucles instables ou des conflits pourraient apparaître. Des stratégies d'**asynchronisme** maîtrisé ou de communication par itérations (steps) successives sont alors recommandées.

Dans cette optique, on peut se référer à diverses **métaphores** ou **règles** d'inspiration biologique. Par exemple, une **mise à jour** "hebbienne" renforce la pondération  $\omega_{i,j}$  lorsqu'on observe une co-activation récurrente des entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ , et l'affaiblit en l'absence de co-occurrence. Par ailleurs, on peut procéder à une **sélection** ou "reproduction" des liaisons : si certaines paires  $(i, j)$  affichent un score ou une utilité trop faible de manière persistante, elles sont éliminées, tandis que d'autres liens se consolident ou se "reproduisent" (création de liaisons analogues sur d'autres nœuds). Une **compétition** locale peut également réguler la densité : on peut contraindre la somme  $\sum_j \omega_{i,j}$  pour chaque entité  $\mathcal{E}_i$  ou recourir à un "softmax" local, privilégiant quelques connexions fortes plutôt qu'un grand nombre de liens faibles.

Par ces variantes, on parvient à **contrôler** la **densité** du graphe, la vitesse de renforcement des pondérations et la **résilience** face au bruit ou aux données incomplètes (sections 1.7.1.2 et 1.5.3). Le **DSL** gagne alors en **scalabilité** tout en maintenant le principe d'**auto-organisation** : chacun de ces mécanismes demeure local, assurant une dynamique distribuée qui s'accorde bien à des mises en œuvre parallèles (GPU, TPU) ou à des approches multi-agents où chaque entité évolue au fil de ses propres observations et interactions.

### 1.7.3.2. Formulation Globale via Fonction d'Énergie

Dans une perspective plus proche de l'**optimisation** (au sens des approches traditionnelles en IA ou en physique statistique), il est concevable de décrire l'**auto-organisation** du Deep Synergy Learning (DSL) par une **fonction d'énergie**  $\mathcal{J}(\Omega)$ . On peut écrire une expression qui généralise la somme des synergies et la pondère par un terme de **régularisation**, selon :

$$\mathcal{J}(\Omega) = - \sum_{i,j} \omega_{i,j} S_{i,j} + R(\Omega).$$

Dans ce formalisme, la somme  $\sum_{i,j} \omega_{i,j} S_{i,j}$  valorise la **coopération** (au sens de la synergie, voir sections 1.4.4 et 1.4.5), tandis que le terme  $R(\Omega)$  constitue une **régularisation**, par exemple  $\alpha \|\Omega\|^2$  ou  $\alpha \|\Omega\|_0$ , qui maintient la parcimonie ou la cohérence structurelle du **Synergistic Connection Network**. Cette formulation unifie les principes évoqués en 1.7.1.4 (contrôle de la densité, maintien de contraintes) dans une unique fonction de coût : l'objectif global consiste à **maximiser** la somme des synergies ou, en termes d'énergie, à **minimiser**  $\mathcal{J}$ .

Pour résoudre l'évolution de  $\Omega$  dans ce cadre, on peut recourir à une **descente de gradient** : si l'on note  $\omega_{i,j}$  un des paramètres, sa dérivée partielle vis-à-vis de  $\mathcal{J}$  prend la forme

$$\frac{\partial \mathcal{J}}{\partial \omega_{i,j}} = -S_{i,j} + \frac{\partial R(\Omega)}{\partial \omega_{i,j}}.$$

En discrétisant cette descente, on retrouve l'équation qui met à jour  $\omega_{i,j}$  de façon incrémentale :

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

ce qui explicite la proximité entre la **logique locale** décrite en 1.4.5 et la **formulation** d'une énergie globale  $\mathcal{J}(\Omega)$ . Dans les problèmes complexes, la descente de gradient peut aboutir à des minima locaux sous-optimaux. On peut alors envisager un **recuit simulé** (simulated annealing) directement sur l'espace du graphe  $\Omega$ . Cette méthode consiste à perturber par moments la configuration courante (création ou suppression de liens), et à n'accepter ou refuser ces modifications qu'avec une probabilité liée à la variation  $\Delta \mathcal{J}$  et à un paramètre de température qui décroît au fil des itérations. Dans le contexte du DSL, où la synergie peut être n-aire (voir la section 1.4.7), cette exploration globale de l'espace combinatoire peut s'avérer fondamentale : la simple descente par ajustements locaux risquerait de rester piégée dans un *cluster* partiel. Le recuit simulé ou des **algorithmes évolutionnaires** jouent alors le rôle de recherche d'une configuration plus robuste, accompagnant l'auto-organisation locale. Cette combinaison de l'approche **par énergie** et de la **dynamique en continu** ouvre une voie pour maîtriser mathématiquement la convergence et la structure du **Synergistic Connection Network**, en particulier lorsqu'on cherche à maintenir une **parsimonie** explicite ou à inclure des contraintes topologiques (voir 1.7.1.4).

### 1.7.3.3. Synergie n-aire et Extensions Conditionnelles

Le **Deep Synergy Learning (DSL)** se focalise souvent sur la **coopération** entre deux entités  $\mathcal{E}_i$  et  $\mathcal{E}_j$ . Toutefois, certains problèmes exigent une **synergie n-aire**, c'est-à-dire impliquant un **groupe**  $\{\mathcal{E}_{i_1}, \dots, \mathcal{E}_{i_n}\}$  de cardinal  $n \geq 3$ . Dans ce cas, on peut introduire des **hyper-arêtes**  $\omega_{i_1, \dots, i_n}$  plutôt que de simples liaisons binaires. La mise à jour de la pondération hyper-arête s'effectue alors selon une généralisation de la loi (voir section 1.4.5), par exemple :

$$\omega_{i_1, \dots, i_n}(t+1) = \omega_{i_1, \dots, i_n}(t) + \eta [S_n(\mathcal{E}_{i_1}, \dots, \mathcal{E}_{i_n}) - \tau \omega_{i_1, \dots, i_n}(t)].$$

Dans cette équation, la fonction  $S_n(\mathcal{E}_{i_1}, \dots, \mathcal{E}_{i_n})$  évalue la **synergie** globale de l'ensemble  $\{\mathcal{E}_{i_k}\}$ . Sa définition peut s'appuyer sur une co-information n-aire, sur un gain de performance collectif ou sur toute métrique reflétant l'apport mutuel de ce sous-groupe. Néanmoins, la **complexité** combinatoire s'en trouve rapidement démultipliée, car on doit gérer des hyper-arêtes potentiellement innombrables lorsque  $n$  augmente. Pour cette raison, il est fréquent de recourir à des **heuristiques** afin de détecter des triplets, quadruplets, etc., seulement lorsque l'on constate une coopération stable entre ces entités ; on se contente de créer l'hyper-arête  $\omega_{i_1, \dots, i_n}$  après avoir observé à maintes reprises que ces  $n$  entités coopèrent ensemble.

Par ailleurs, il arrive que la **synergie** dépende d'un **contexte**  $\mathbf{c}$ . Un exemple typique survient lorsque l'association audio-texte n'est pertinente que dans une situation donnée (un lieu, une heure, une ambiance sonore). La règle de mise à jour se réécrit alors :

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

Dans ce cadre, on doit calculer un  $S(\cdot \mid \mathbf{c})$  pouvant fortement varier en fonction du contexte. La tâche d'**estimation** et d'**optimisation** se complexifie davantage : la distribution induite par  $\mathbf{c}(t)$  peut changer au fil du temps ou comprendre divers régimes de fonctionnement. Les algorithmes d'auto-organisation du DSL doivent alors incorporer des modèles de synergie **conditionnelle**, qui modulent la pondération  $\omega_{i,j}$  suivant la valeur prise par  $\mathbf{c}$ . Cette extension demeure essentielle dans de multiples applications, comme la robotique adaptative (un même couple de capteurs se révèle pertinent ou non selon le mode de locomotion) ou le traitement multimodal (un segment audio et un

mot ne coïncident que dans un contexte temporel précis). Elle requiert cependant des stratégies de **parcimonie** ou de **sampling** pour éviter une explosion combinatoire lorsqu'on veut évaluer la synergie conditionnelle sur tous les contextes possibles.

#### 1.7.3.4. Contrôle de la Stabilité et Convergence Globale

Dans tout **système dynamique** présentant un large couplage entre ses éléments, il est fréquent d'observer des **cycles** ou des **oscillations** : un sous-réseau se renforce pendant un temps, puis un autre prend le relais en affaiblissant les premiers liens, et ainsi de suite. Il peut aussi survenir un phénomène de **basculement** entre plusieurs attracteurs stables (configurations rivales) si la structure du **Synergistic Connection Network** autorise plusieurs manières de s'organiser pour la même situation. Le **Deep Synergy Learning (DSL)** n'échappe pas à ces scénarios, en particulier lorsque la fonction de synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$  est très "non linéaire" ou quand la synergie n-aire entre plusieurs entités génère des interactions complexes (sections 1.4.7 et 1.7.3.3).

Pour limiter ce risque d'oscillations ou de dérive, il convient d'inclure des **mécanismes** ou des **règles** de contrôle dans la mise à jour des pondérations  $\omega_{i,j}$ . On peut imposer un **amortissement** progressif (faire varier  $\eta$  ou  $\tau$  dans le temps), intégrer une **pénalisation** plus agressive des liens excessivement grands, ou encore définir des **seuils de saturation** empêchant une pondération  $\omega_{i,j}$  de croître sans borne. Des analyses mathématiques, telles que l'étude de la **Jacobienne** locale ou des **fonctions de Lyapunov** globales, peuvent démontrer qu'un schéma de mise à jour particulier garantit la convergence vers un état stable (ou au moins vers un cycle stable), sans risquer un chaos indésirable.

Il est également envisageable de greffer un **module** de contrôle hiérarchique (de plus haut niveau), qui surveille la dynamique globale du réseau et injecte ponctuellement des modifications de paramètres  $\eta$  ou  $\tau$  si la variance du système dépasse un certain seuil. Ce principe d'**hybridation** (auto-organisation libre + régulation macro) rappelle certaines structures biologiques ou cognitives, où les ajustements locaux cohabitent avec des boucles de rétroaction plus centrales. On cherche alors à préserver la **liberté synergique** au niveau local, tout en s'assurant que le **réseau** ne sombre pas dans une instabilité excessive. Cet équilibre soulève des questions d'**optimisation** et de **pilotage** du **DSL**, qui restent ouvertes à la recherche, notamment sur les plans pratique (implémentation à grande échelle) et théorique (analyse de la dynamique dans un espace de forte dimension).

### Conclusion

Le **développement d'algorithmes d'optimisation** pour le **Deep Synergy Learning** s'annonce particulièrement riche et complexe. Les **règles de mise à jour** naïves suffisent à illustrer le concept, mais ne passent pas toujours à l'échelle ni ne gèrent la synergie n-aire, les contraintes symboliques, ou la minimisation globale d'une fonction d'énergie. Les approches possibles incluent :

- **Mise à jour locale** inspirée de la biologie (sélection, Hebb, normalisation compétitive),
- **Descente d'énergie** globale (fonctions  $\mathcal{J}$  pénalisant la densité, favorisant la synergie),
- **Algorithmes évolutifs** (recuit simulé, heuristiques stochastiques) pour éviter les minima locaux,
- **Extensions** aux hyper-arêtes (synergies n-aires), aux synergies conditionnelles, ou aux règles symboliques,
- **Contrôles** de stabilité et d'amortissement, voire un superviseur partiel pour limiter les oscillations et injecter des priorités.

Trouver la meilleure **formulation** et le meilleur **algorithme** dépendra de l'application (vision, robotique, recommandation, etc.), du **volume** de données et de la **structure** (binaire ou n-aire, symbolique, multi-modal). La poursuite de ces recherches est incontournable pour que le **DSL** gagne en robustesse, en efficacité, et en adoptabilité dans le monde réel.

#### 1.7.4. Contrôle et Stabilité des Processus Auto-Organisés

L'**auto-organisation** est au cœur du **Deep Synergy Learning (DSL)**, permettant à un grand nombre d'entités (capteurs, modules de calcul, représentations symboliques ou sub-symboliques) de s'agencer spontanément en **clusters** ou **macro-clusters** selon leurs synergies. Cette dynamique, tout en conférant au DSL sa **plasticité** et sa **capacité d'adaptation**, peut aussi engendrer des **problèmes de stabilité** : oscillations, attracteurs multiples, comportements chaotiques, etc. De plus, dans bien des cas (robotique, diagnostic, logistique), il faut **contrôler** partiellement la configuration auto-organisée, ne serait-ce que pour garantir la **sécurité**, la **fiabilité**, ou la **cohérence** avec des règles externes.

Dans cette sous-section (1.7.4), nous examinons les **défis** posés par la **dynamique** d'un système DSL, et les **solutions** ou **mécanismes** possibles pour assurer un certain niveau de **contrôle** et de **stabilité** :

58. La tendance des systèmes couplés à générer des **oscillations** ou des **cycles**,

59. Les risques d'**explosion** ou d'**effondrement** des pondérations synergiques  $\omega$ ,
60. L'existence de **multiples attracteurs** en concurrence, pouvant conduire à des configurations divergentes,
61. Les **dispositifs** de rétroaction ou de surveillance globale pour introduire un pilotage **hiérarchique** léger,
62. Les **garanties** (ou non-garanties) de convergence sous certaines hypothèses mathématiques.

#### 1.7.4.1. Risques d'Oscillations et de Comportements Chaotiques

Dans le **Deep Synergy Learning**, chaque liaison  $\omega_{i,j}$  obéit à une évolution itérative inspirée des principes décrits en section 1.4.5, ce qui donne lieu à un **système dynamique** d'environ  $O(n^2)$  degrés de liberté dans le cas binaire. Pour comprendre comment apparaissent des **cycles** ou des **régimes oscillatoires**, il est éclairant d'étudier la **linéarisation** locale autour d'un point fixe  $\omega^*$ . Supposons l'existence d'un équilibre  $\omega^*$  satisfaisant un état stationnaire ; si l'on introduit une petite perturbation  $\delta\omega$ , la dynamique globale s'écrit de manière approchée :

$$\delta\omega(t+1) \approx J(\omega^*) \delta\omega(t),$$

où  $J(\omega^*)$  désigne la **matrice jacobienne** évaluée en  $\omega^*$ . En temps discret, un comportement oscillatoire ou pseudo-chaotique survient lorsque certaines **valeurs propres** de  $J$  possèdent une magnitude supérieure à 1. Cela signifie que la perturbation  $\delta\omega$  se voit **amplifiée** au fil des itérations, au lieu d'être dissipée. En temps continu, la condition équivalente consisterait à observer des **parties réelles** positives dans les valeurs propres, menant à une instabilité locale ou à des boucles auto-entretenues.

Ce type de phénomène peut se traduire par des **cycles** permanents, ou même des trajectoires quasi chaotiques, contrariant la **convergence** du réseau vers un état stable. De tels régimes, bien que mathématiquement fascinants, s'avèrent peu souhaitables si l'on souhaite une structure DSL stable pour la robotique, la maintenance ou la prise de décision. C'est pourquoi on introduit souvent des **mécanismes** de stabilisation. Une première solution consiste à contrôler la "vitesse" de mise à jour en imposant une contrainte, par exemple  $\eta \tau < 1$ , dans le cas linéaire simplifié. On peut ensuite recourir à des **termes** non linéaires de saturation, destinés à éviter la croissance illimitée de certaines pondérations. Un exemple simple, dans un modèle continu, consiste à enrichir l'équation d'évolution par un terme  $-\beta (\omega_{i,j})^3$  :

$$\frac{d}{dt} \omega_{i,j} = \eta [S(\mathcal{E}_i, \mathcal{E}_j) - \tau \omega_{i,j}] - \beta (\omega_{i,j})^3,$$

où  $\beta > 0$  modère la croissance des liaisons fortes. De manière analogue, on peut introduire un **facteur** de compétition locale (type softmax), en imposant  $\sum_j \omega_{i,j} \leq K$  pour chaque entité  $\mathcal{E}_i$ , de sorte qu'une entité ne puisse pas développer des liens solides avec trop de partenaires en même temps.

Ces mesures limitent la **liberté** d'auto-organisation, mais s'avèrent cruciales pour écarter des régimes oscillatoires ou chaotiques susceptibles d'émerger dans un réseau fortement couplé et non linéaire. Elles assurent ainsi une forme de **stabilité** globale tout en préservant la possibilité de réorganisations locales dictées par la synergie. Un savant réglage des paramètres ( $\eta$ ,  $\tau$ ,  $\beta$ , ou le budget  $\sum_j \omega_{i,j} \leq K$ ) permet donc de maintenir le **DSL** dans une zone où le réseau demeure robuste et apte à converger, évitant des fluctuations intempestives tout en conservant l'essence auto-adaptative voulue.

#### 1.7.4.2. Multiples Attracteurs, Convergence Incertaine

Il est fréquent qu'un **Deep Synergy Learning (DSL)** cherche à maximiser, ou à minimiser l'opposé, d'une **somme de synergies**  $\sum_{i,j} \omega_{i,j} S_{i,j}$  éventuellement modulée par un terme de pénalisation ou de parcimonie (voir section 1.7.1.4). Dans un espace de grande dimension, on peut rencontrer de **multiples attracteurs** correspondant à des configurations dont l'énergie (ou le coût) est quasi équivalente. Ce phénomène se manifeste lorsqu'il existe plusieurs manières d'**arranger** les entités en clusters, chacune procurant un gain global proche. Le réseau peut alors converger vers l'un ou l'autre de ces attracteurs selon les conditions initiales ou de légères fluctuations.

Dans la pratique, il arrive que des "sous-réseaux" rivaux émergent au cours de la dynamique, chacun rassemblant certaines entités en un **cluster** distinct. Une compétition peut alors s'installer entre ces sous-réseaux, induisant des **instabilités** transitoires et des oscillations (voir section 1.7.4.1). Si l'on désire un **état** unique et stable, par exemple pour un robot qui ne peut se scinder en deux stratégies opposées, on peut injecter un **signal global** orientant le réseau vers l'attracteur souhaité. Cette intervention se formalise en modifiant la fonction de synergie ou en ajoutant un **feedback** de récompense plus élevé pour une configuration A que pour une configuration B. On obtient ainsi un **pilotage** macro qui restreint la pure auto-organisation, mais assure qu'en présence de minima multiples, la configuration la plus souhaitable selon les critères d'usage devienne prépondérante. Cette démarche montre le besoin de compromis entre la liberté synergique propre au DSL et la nécessité d'un **contrôle** hiérarchique dans certaines applications critiques.

### 1.7.4.3. Approches Hiérarchiques Légères pour Guider l'Auto-Organisation

Dans des environnements critiques, tels que le domaine **médical** ou les **transports**, il n'est pas toujours acceptable de laisser un réseau en **Deep Synergy Learning (DSL)** se reconfigurer librement sans un regard d'ensemble. Afin de maîtriser la sécurité et la fiabilité, on introduit souvent une **couche** ou un **module** de supervision qui contrôle globalement la structure  $\Omega$ . Ce superviseur observe la **matrice** des liens actifs et vérifie des **contraintes** imposées par l'application. Par exemple, on peut exiger que la somme  $\sum_j \omega_{i,j}$  attachée à une entité  $\mathcal{E}_i$  ne dépasse pas un seuil spécifique, ou que certaines entités déclarées incompatibles (règles de sécurité, normes médicales contradictoires) ne figurent pas dans le même cluster. Si ce superviseur détecte une violation, il peut forcer la **mise à zéro** de certaines liaisons  $\omega_{i,j}$ , neutralisant ainsi des configurations illégitimes.

On peut aussi concevoir un **schéma** où la **couche basse** demeure l'auto-organisation locale et distribuée, tandis qu'une **couche haute** (un planificateur ou un décideur global) injecte à intervalles réguliers des **influx** spécifiques dans le réseau. Ces influx peuvent prendre la forme d'une modification ponctuelle de  $\omega_{i,j}$ , d'un renforcement sélectif d'une synergie, ou de l'assignation de priorités à certaines entités. Cet apport d'information agit comme une **guidance** de haut niveau, combinée à la dynamique adaptative du DSL.

D'un point de vue mathématique, on peut l'interpréter comme l'ajout d'un **terme** exogène dans l'équation de mise à jour, par exemple  $\Delta\omega_{i,j}^{(\text{macro})}$  reflétant la volonté du **planificateur**. Cela permet d'enrichir la structure  $\Omega$  sans remettre en cause l'intégralité du principe d'**auto-organisation** locale. On obtient un **compromis** dans lequel la dynamique  $\omega_{i,j}$  se déroule principalement selon la synergie perçue, tout en restant arrimée à des objectifs de **performance** ou de **sécurité** plus vastes. Dans ce type d'architecture dite « hybride », le contrôle hiérarchique léger se borne à préserver la cohérence, tandis que la plupart des adaptations opérationnelles sont gérées par la mise à jour distribuée des liaisons.

### 1.7.4.4. Outils Mathématiques et Théoriques pour la Stabilité

Il est souvent utile de modéliser l'**évolution** d'un **Deep Synergy Learning (DSL)** sous la forme d'un **système dynamique**  $\omega(t+1) = \omega(t) + F(\omega(t))$ . Dans cette notation,  $\omega(t)$  regroupe l'ensemble des pondérations  $\omega_{i,j}(t)$ , et  $F(\omega(t))$  décrit la contribution que la synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$  (ainsi que les termes de décroissance, de régularisation, etc.) apporte à la variation entre  $t$  et  $t+1$ .

Pour **étudier** la stabilité locale autour d'un **point fixe**  $\omega^*$  (ou d'une configuration fixe du réseau), on réalise classiquement une **linéarisation** de  $F$ . Concrètement, on calcule la **matrice jacobienne** :

$$J(\omega^*) = \frac{\partial F}{\partial \omega}(\omega^*).$$

Si l'on note  $\delta\omega(t) = \omega(t) - \omega^*$ , alors pour de petites perturbations, on obtient un modèle linéarisé :

$$\delta\omega(t+1) \approx \delta\omega(t) + J(\omega^*) \delta\omega(t).$$

En régime de temps discret, la **stabilité** locale autour de  $\omega^*$  se produit si toutes les **valeurs propres** de  $I + J(\omega^*)$  ont une magnitude strictement inférieure à 1. Dans un cadre continu (où l'on écrit  $d/dt \omega = F(\omega)$ ), il faut que toutes les parties réelles des valeurs propres de  $J(\omega^*)$  soient **négatives**. Cette condition s'interprète physiquement : de petites déviations  $\delta\omega$  s'amortissent au fil du temps. Elle dépend du **choix** de la synergie  $S(\mathcal{E}_i, \mathcal{E}_j)$ , de la taille du pas  $\eta$ , des paramètres de décroissance  $\tau$ , ou des non-linéarités supplémentaires (termes de saturation, etc.).

Parfois, on peut exhiber une **fonction de Lyapunov**  $\Phi(\omega)$  qui décroît strictement à chaque mise à jour, prouvant la **convergence** vers un **unique attracteur**. Cela exige que la définition de la synergie ne comporte pas de **non-linéarités** trop complexes, ou que la structure ne s'étende pas à des hyper-arêtes de dimension élevée (voir la section 1.4.7 sur la synergie n-aire). Dans des **configurations** plus simples (synergie linéarisable, réseau symétrique, etc.), on peut démontrer rigoureusement l'existence et l'unicité d'un état stationnaire stable. En revanche, pour des **formulations** plus riches (synergie hautement non linéaire, dépendance au contexte, hyper-arêtes n-aires), établir une **preuve** de stabilité globale devient complexe : la dynamique peut potentiellement présenter de multiples attracteurs ou des régimes oscillants.

Ces outils (linéarisation jacobienne, fonctions de Lyapunov) offrent néanmoins un **cadre** théorique solide pour analyser la mise à jour des pondérations  $\omega_{i,j}$  dans un **DSL**, en particulier lorsque l'on cherche à garantir un comportement prédictible et éviter les **oscillations** chaotiques ou l'explosion des liaisons. Ils fournissent aussi des guides de **conception** : on peut paramétrer  $\eta$  et  $\tau$  ou imposer des contraintes de somme, de norme, etc. afin de maintenir la trajectoire de  $\omega$  à l'intérieur d'une **région** stable et de faire en sorte que le système converge réellement, au lieu de dériver ou de switcher entre multiples configurations.

## Conclusion

Assurer le **contrôle** et la **stabilité** des **processus auto-organisés** dans le **DSL** est un **enjeu fondamental** pour :

- Prévenir l'apparition de **comportements oscillants** ou chaotiques,
- Gérer la coexistence de **multiples attracteurs** et choisir (ou influencer) l'état final,
- Introduire un **encadrement** (module de supervision, règles hiérarchiques minimales) pour les systèmes critiques ou lorsqu'on souhaite une trajectoire de réorganisation déterministe.

Les **solutions** mathématiques ou algorithmiques incluent :

- La **conception** de règles de mise à jour (amortissement, saturation, normalisation) qui limitent la croissance excessive des pondérations,
- L'introduction d'une **énergie globale**  $\mathcal{J}(\Omega)$ , encadrée par des méthodes de descente ou de recuit simulé,
- Des **mécanismes** de **pilotage macro** (modification de  $\eta$ ,  $\tau$ , injection de priorités, filtration de certaines liaisons) dans un schéma hybride auto-organisé + contrôle haut-niveau,
- Des **analyses** de stabilité via Jacobienne locale ou Lyapunov, pour garantir (au moins localement) la convergence ou la non-explosion.

En somme, maîtriser l'**auto-organisation** ne signifie pas brider la flexibilité du DSL, mais offrir à l'architecture la **robustesse** nécessaire pour opérer en conditions réelles, où l'on ne peut se permettre un comportement erratique ni une multiplicité de configurations irréciliables.

#### 1.7.5. Interprétabilité et Explicabilité pour l'Humain

Un objectif crucial dans la plupart des applications d'**intelligence artificielle** est de fournir non seulement des **résultats** (prédictions, recommandations, décisions), mais également une **compréhension** de la logique ou du **processus** qui y aboutit. Les méthodes classiques de **Deep Learning** (CNN, RNN, Transformers) sont souvent décrites pour leur caractère "**boîte noire**", dans la mesure où l'information se diffuse à travers des poids distribués dans de multiples couches, rendant l'**explication** pour l'humain délicate. Le **Deep Synergy Learning (DSL)**, de par sa **structure** en **entités** et en **liens synergiques** ( $\omega_{i,j}$ ), fournit un cadre potentiellement plus **transparent** : on peut visualiser **quelles** entités interagissent fortement, **comment** se constituent les **clusters** ou **macro-clusters**, et **pourquoi** ces regroupements émergent (sur la base d'une synergie mesurable). Toutefois, obtenir une **interprétabilité** réelle et exhaustive n'est pas garanti : il faut des **mécanismes** méthodologiques pour extraire des **explications** compréhensibles. Cette sous-section (1.7.5) discute les **défis** et **pistes** liées à l'explicabilité dans le DSL.

##### 1.7.5.1. D'un Modèle Hiérarchique Opaque à un Réseau de Liens Synergiques

Il est courant que les **réseaux neuronaux profonds** (CNN, RNN, Transformers) diluent leur représentation interne à travers plusieurs **couches** qui transforment graduellement les descripteurs, rendant ainsi les décisions finales difficiles à expliquer. Les neurones internes n'ont pas d'**identité** propre : ils sont simples réceptacles de poids et d'activations, et il est souvent ardu d'y associer un sens direct. À l'inverse, le **Deep Synergy Learning (DSL)** introduit un **réseau** où chaque **nœud**  $\mathcal{E}_i$  possède une **identité** clairement définie (il peut s'agir d'un **capteur** particulier, d'un **concept** sémantique, d'un **patch** visuel, d'une **règle** symbolique, etc.), tandis que les **liaisons**  $\omega_{i,j}$  se **reconfigurent** progressivement, au gré de la synergie détectée entre  $\mathcal{E}_i$  et  $\mathcal{E}_j$ .

Cette structure procure plusieurs **avantages**. Elle favorise d'abord la **lisibilité** de certains **sous-ensembles** : on peut dénicher des **clusters** (chapitre 1.4.3) formés par un groupe d'entités entretenant des liens forts, et y voir immédiatement une **cohérence** (par exemple, un ensemble "visuel + audio" en reconnaissance multimodale ou un ensemble "utilisateur + contenus + tag de genre" en recommandation). Par ailleurs, chaque **entité**  $\mathcal{E}_i$  se voit attribuer une **signification** (un patch d'image, un concept symbolique), ce qui facilite grandement l'**explication** du raisonnement ou de la recommandation. Un **macro-cluster** peut ainsi refléter un groupement plus vaste : "groupe d'utilisateurs amateurs de jazz + contenus jazz + soirées concert" dans une plateforme culturelle, offrant une justification naturelle à la recommandation "puisque vous faites partie du cluster jazz, on vous propose ce concert".

Cependant, cette **lisibilité** accrue sur le plan conceptuel s'accompagne aussi d'une **complexité** potentiellement élevée. Le réseau DSL peut afficher un grand nombre de **nœuds** (surtout en contexte industriel ou multi-sensoriel), et chaque nœud peut avoir plusieurs **connexions** significatives, aboutissant à une **topologie** complexe, souvent composée de **clusters** imbriqués ou de multiples liens réciproques. Une simple **visualisation** naïve des poids  $\omega_{i,j}$  peut alors ne rien révéler de clair, s'il n'existe pas de mécanisme de filtrage ou de simplification. Il devient nécessaire d'utiliser des **outils** de **hiérarchisation**, de **regroupement** ou de **filtrage** pour dériver des **chemins** explicatifs, par exemple en extrayant les principales **chaînes** de liaison qui influencent la décision finale. C'est là que réside l'enjeu d'une **explicabilité** réellement transparente dans un réseau DSL riche, qui exige de se doter de techniques aptes à condenser le graphe et à exposer les assemblées cruciales, tout en évinçant la masse de connexions moins pertinentes.



### 1.7.5.2. Extractions de Chemins Synergiques et Clusters Pivot

Dans un **réseau** construit selon les principes du **Deep Synergy Learning (DSL)**, il est possible d'expliquer une décision ou une recommandation en identifiant un **chemin** significatif reliant deux entités  $\mathcal{E}_a$  et  $\mathcal{E}_b$ . On peut ainsi considérer une suite :

$$\mathcal{E}_a \rightarrow \mathcal{E}_x \rightarrow \mathcal{E}_y \rightarrow \mathcal{E}_b,$$

au long de laquelle la **synergie** s'avère forte. On peut évaluer cette force par un produit du type  $\omega_{a,x} \omega_{x,y} \omega_{y,b}$ , exprimant la continuité du lien depuis  $\mathcal{E}_a$  jusqu'à  $\mathcal{E}_b$  via  $\mathcal{E}_x$  et  $\mathcal{E}_y$ . Par exemple, si  $\mathcal{E}_a$  est un **utilisateur** et  $\mathcal{E}_b$  un **contenu** recommandé, un **chemin** pertinent pourrait comporter un **tag** musical  $\mathcal{E}_x$  et un **artiste**  $\mathcal{E}_y$  : la synergie souligne alors que l'utilisateur  $\mathcal{E}_a$  est historiquement associé à ce tag rock, lequel est fortement relié à l'artiste  $\mathcal{E}_y$ , et que cet artiste conduit au **contenu**  $\mathcal{E}_b$ . L'explication se formule sous la forme d'un **raisonnement local** : "Vous êtes lié à ce tag rock, ce tag rock est fortement corrélé à l'artiste  $\mathcal{E}_y$ , et cet artiste conduit au contenu  $\mathcal{E}_b$  proposé." Dans la pratique, on limite souvent la longueur du **chemin** afin de conserver une **explication** concise et compréhensible, tout en montrant comment la synergie chemine dans le **Synergistic Connection Network**.

Une autre forme d'explication s'appuie sur le **cluster** (ou **macro-cluster**) dans lequel réside l'entité  $\mathcal{E}_b$  qu'on veut justifier. On expose alors les **entités dominantes** de ce cluster : par exemple, en diagnostic médical, on peut lister certains **indicateurs** (fièvre, anomalie sanguine, image radio suspecte) qui forment ensemble un sous-réseau hautement interconnecté. L'**interprétation** émerge alors de la **cohérence interne** du **cluster** : "Ces signes, fortement liés par la synergie, corroborent la conclusion de pathologie." Cette méthode est très usitée lorsque la configuration ne s'explique pas par un unique chemin, mais par un **ensemble** d'interactions locales qui se cristallisent en un groupement solidaire. L'analyse du cluster met en avant les entités centrales et leur degré de liaison  $\omega$ , rendant la décision plus **transparente**.

### 1.7.5.3. Couplage Symbolique pour des Justifications de Haut Niveau

Dans la section 1.5.7, il a été expliqué comment intégrer des **règles** ou des **concepts** symboliques au sein du **Deep Synergy Learning (DSL)** de manière à fusionner des entités strictement sub-symboliques (features perceptuelles, segments, etc.) et des entités logico-sémantiques. Cette intégration prend toute son importance pour l'**explicabilité**, car elle autorise la production de justifications directement lisibles, sans nécessiter un métamodèle séparé. Si l'on imagine une **règle**  $\mathcal{E}_{rule}$  affirmant "cercle + chiffre => panneau de limitation", la présence d'un lien fort  $\omega_{rule, visCircle}$  met en évidence la façon dont la règle coopère avec l'entité  $\mathcal{E}_{visCircle}$  qui détecte les formes circulaires, et un autre lien  $\omega_{rule, txt30}$  connecte cette règle à l'entité textuelle représentant le chiffre "30". La combinaison de ces liaisons convainc ainsi que la **reconnaissance** du panneau 30 km/h découle d'une synergie tangible entre la règle symbolique "cercle + chiffre" et les observations sub-symboliques qui valident les deux prérequis.

Cette **cohabitation** symbolique-subsymbolique s'avère donc profitable pour expliquer pourquoi une règle s'active, de quelle façon elle est validée par la scène visuelle ou contextuelle, et comment cette coopération se traduit en une conclusion. Le réseau **DSL**, formé de pondérations  $\{\omega_{i,j}\}$ , expose explicitement la manière dont la règle " $\mathcal{E}_{rule}$ " tisse des liens avec des features visuelles ou auditives pour générer une reconnaissance de haut niveau. Il est en outre possible d'**extraire** des proto-règles depuis les clusters émergents dans le réseau : si l'on observe qu'un **cluster** demeure très stable, rassemblant certaines variables (par exemple, plusieurs marqueurs biologiques), un symptôme et un diagnostic, on peut reformuler une **règle émergente** indiquant que "lorsque X, Y, Z sont élevés, la synergie avec le diagnostic D augmente considérablement, ce qui équivaut à suspecter telle pathologie P." Cette forme d'**abstraction** dérive naturellement de la dynamique d'auto-organisation du **DSL** et fournit une explication concise, en reliant directement le diagnostic à un sous-réseau stable de variables médicales fortement pondérées. On franchit ainsi une étape supplémentaire vers une **explicabilité** plus intuitive, où l'on ne se contente pas de mentionner le rôle de quelques features, mais où l'on propose une **règle** ou une **raison** qui s'apparente à un énoncé symbolique, ancré dans les liaisons sub-symboliques du réseau.

### 1.7.5.4. Points de Vigilance : Graphes Trop Grands, Hétérogénéité Massive

Dans un **Deep Synergy Learning** de forte dimension, la **visualisation** intégrale de toutes les liaisons  $\omega_{i,j}$  ou l'exploration exhaustive des chemins reliant deux entités devient invisable. La taille du graphe peut atteindre plusieurs milliers ou millions de nœuds et de liens, rendant toute inspection manuelle impossible. Afin de surmonter ces limites, il est utile de mettre en place des **mécanismes de filtrage** ou d'**agrégation**. Le filtrage consiste à écarter les liaisons dont la pondération  $\omega_{i,j}$  reste en deçà d'un **seuil** minimal, ou qui ne participent pas de manière significative à un **cluster** pivot. Cette méthode abaisse drastiquement la densité du réseau et en facilite l'interprétation. L'agrégation, quant à elle, regroupe certaines entités en **macro-nœuds**, de sorte que l'on obtienne un **graphe résumé** nettement plus petit, faisant émerger les **clusters** importants tout en préservant la cohérence des liens principaux.

Lorsque les entités du DSL recouvrent une **hétérogénéité** massive, intégrant par exemple de la vision, de l'audio, du texte et des modules symboliques, la **compréhension humaine** impose de clarifier la correspondance entre chaque entité et les données originales. Un patch d'image doit être décrit par sa position ou sa nature dans la scène, un segment audio doit renvoyer à la phrase ou au moment précis de l'enregistrement, un concept symbolique doit être relié à son énoncé explicite. Cette **traduction** partielle s'avère cruciale pour délivrer

une **explicabilité** satisfaisante : l'utilisateur comprend alors pourquoi telle liaison  $\omega_{i,j}$  est forte, parce qu'un patch visuel repéré comme "zone X" s'avère synergiquement lié à un certain concept ou à un segment audio. Il est donc indispensable d'associer à chaque entité  $\mathcal{E}_i$  un ensemble de **méta-informations** décrivant son rôle, son lien avec l'input brut et sa typologie (vision, texte, règle symbolique). Grâce à ce balisage, la **mise en récit** du réseau DSL, même de grande taille, peut s'effectuer de façon plus synthétique et plus pertinente, en se concentrant sur un sous-ensemble restreint d'arêtes fortement pondérées et sur des macro-clusters résumant la structure globale.

## Conclusion

La **structure** en entités et liaisons synergiques du **Deep Synergy Learning** confère un **avantage** potentiel pour l'**explicabilité** : on peut :

- **Identifier** des clusters,
- **Exhiber** des sentiers  $\mathcal{E}_i \rightarrow \mathcal{E}_j$  hautement pondérés,
- **Mettre** en avant des règles symboliques si présentes,
- **Comparer** des liens forts entre un utilisateur/contenu (dans le cas de recommandation) ou un patient/diagnostic (dans le cas médical).

Cependant, l'**explosion** du nombre d'entités et de liens peut rendre difficile la simple "lecture" de la configuration globale. On doit alors s'équiper de **techniques** de filtrage, d'**agrégation** (macro-clusters), ou de **génération** d'explications locales (chemins courts). La cohabitation possible du **symbolique** et du **sub-symbolique** aide à concrétiser des **justifications** plus claires. Ainsi, le DSL peut jouer un rôle dans la mouvance d'**Explainable AI**, à condition de manier soigneusement la **complexité** et la **variété** des entités afin de délivrer une **vision** cohérente et compréhensible à l'utilisateur final.

### 1.7.6. Considérations Éthiques et Réglementaires

Au-delà des enjeux techniques (scalabilité, stabilité, explicabilité), le **Deep Synergy Learning (DSL)** soulève également des questions **éthiques** et **réglementaires**. En effet, l'auto-organisation des entités au sein d'un **réseau** qui évolue sans hiérarchie prédéfinie peut introduire une forme d'**imprévisibilité** ou de **décision distribuée**. Lorsqu'il s'agit de systèmes critiques (médical, financier, militaire, etc.), il est indispensable de **responsabiliser** la conception et l'usage de tels modèles. De même, dans des applications plus courantes (recommandation, transport, logistique), la manière dont les données sont exploitées et dont les clusters se forment peut produire des **biais** ou altérer la **transparence** vis-à-vis des utilisateurs. Cette sous-section (1.7.6) examine :

63. Le **risque** de perte de contrôle ou d'incompréhension du comportement d'un système auto-organisé,
64. Les **biais** possibles et la nécessité de garanties contre la discrimination ou l'iniquité,
65. Les **règles** et normes (RGPD, réglementations sectorielles) exigeant traçabilité et protection des données,
66. Le rôle de **valeurs humaines** (sécurité, dignité, respect de la vie privée) dans la conception de DSL responsables.

#### 1.7.6.1. Perte de Contrôle et Responsabilités

L'un des attraits majeurs du **Deep Synergy Learning (DSL)** réside dans sa **dynamique** d'auto-organisation : les entités s'y regroupent en **clusters** suivant les synergies détectées, sans requérir un programme explicite gérant leurs interactions. Toutefois, cette souplesse peut engendrer des **comportements** difficiles à anticiper, dans la mesure où ni le concepteur, ni l'opérateur, ni le développeur n'ont directement programmé la structure finale. Dans des contextes où la **responsabilité** est cruciale, notamment en milieu médical ou bancaire, il importe de savoir **qui** assumera les conséquences si le système commet un choix discutable ou s'il engendre un dommage. Avec un réseau neuronal "classique", déjà, l'**opacité** rend l'attribution de responsabilité délicate ; dans un DSL à auto-organisation plus distribuée, le degré de complexité est plus élevé encore.

Pour pallier cette difficulté, il est envisageable de munir le système d'un **journal** d'événements et de mises à jour, permettant de **tracer** l'historique de la pondération  $\omega_{i,j}$ . Théoriquement, on pourrait reconstituer le "chemin causal" entre une observation et une décision, même au sein d'un graphe évolutif. Cette traçabilité demeure toutefois ardue à mettre en œuvre si l'on veut documenter en continu un réseau de grande dimension.

La loi ou les **régulations** sectorielles (ex. aéronautique, médical, bancaire) imposent souvent de **certifier** une partie du comportement du système avant son déploiement. Un DSL auto-organisé risque de "**changer**" de configuration quasi à chaque itération, rendant hasardeuse toute garantie statique. Il existe alors plusieurs stratégies. Il est possible de borner la **vitesse** d'évolution (en limitant le produit  $\eta \times \tau$ , ou



en imposant un pas maximal) pour qu'il demeure un **voisinage** de configurations déjà validées. On peut aussi intégrer un **module** de supervision hiérarchique (voir la section 1.7.4.3) qui invalide certaines reconfigurations jugées non conformes à des normes ou des plans de sûreté. Enfin, il est envisageable d'appliquer des **tests** de robustesse, ou "stress tests", simulant divers scénarios de données pour s'assurer qu'aucune configuration délétère ne surgisse, ou du moins qu'elle reste suffisamment rare et détectable.

Ces considérations reflètent un compromis entre la **liberté** synergique, qui fait la force conceptuelle du DSL, et la **maîtrise** requise dans les secteurs critiques. Les débats sur la responsabilité sont dès lors analogues à ceux suscités par les réseaux neuronaux opacifiés, mais se complexifient en raison de la reconfiguration continue du réseau. Les utilisateurs, exploitants et autorités de tutelle doivent tenir compte de la difficulté d'isoler un instantané du DSL pour en vérifier la validité et assumer les conséquences de ses décisions évolutives.

#### 1.7.6.2. Biais, Équité et Discrimination

Dans un **Deep Synergy Learning (DSL)** appliqué à des tâches de recommandation ou de décision, il arrive que la dynamique de formation des **clusters** reflète des **biais** préexistants dans les données, ce qui peut engendrer des effets discriminants. Imaginons un système de **scoring** pour l'attribution de crédits bancaires. Si les bases de données d'emprunteurs présentent une corrélation injustifiée entre certaines variables socio-économiques et la capacité de remboursement, le **DSL** risque de regrouper un certain sous-groupe dans un **cluster** jugé risqué, ou de l'isoler dans un espace périphérique, renforçant une forme d'exclusion. Ces liaisons faibles ou inexistantes maintiennent le sous-groupe à l'écart des opportunités, ce qui peut amplifier la ségrégation : moins on l'autorise à coopérer avec le reste du réseau, plus sa synergie chute, et moins il bénéficie d'un accès équitable au crédit.

Ce phénomène d'**auto-renforcement** est potentiellement dangereux. Dans un simple réseau neuronal supervisé, des régulations ou des contraintes peuvent être imposées pour limiter l'exploitation de variables sensibles. Mais dans un **DSL**, la situation est plus complexe, car les **entités** se lient ou se séparent selon la synergie globale perçue. Un sous-groupe minoritaire, s'il n'est pas bien représenté dans les données, peut se voir marginalisé. Il convient alors d'établir des **mécanismes** préventifs ou correctifs. Il est envisageable, par exemple, de **pénaliser** l'isolation injustifiée d'un sous-groupe, en introduisant un terme additionnel dans la fonction d'énergie ou dans l'équation de mise à jour des pondérations. Ce terme peut imposer qu'un certain ensemble d'entités (associées à un attribut protégé) ne soit pas trop "distant" du reste ou qu'un cluster ne se forme pas exclusivement autour d'une caractéristique stigmatisante. Un autre moyen consiste à **rééquilibrer** les données avant l'apprentissage, afin de fournir au DSL une image plus fidèle et plus inclusive de la population, ou à injecter artificiellement des entités compensatoires, assurant que le réseau aura de quoi évaluer la synergie de manière équitable.

Dans un cadre légal, des obligations comme le **RGPD** (Règlement Général sur la Protection des Données) ou les lois anti-discrimination exigent que la décision ne dépende pas d'informations relatives à l'origine ethnique, à la religion ou à d'autres critères protégés. Le **DSL**, en créant des entités et des liaisons auto-organisées, doit donc s'assurer que certaines variables n'encoderont pas implicitement ces attributs sensibles, sous peine d'aboutir à des comportements discriminants difficilement décelables. Il est crucial de mettre en place des **audits** récurrents du réseau pour vérifier la distribution des **clusters**, observer si un groupe spécifique demeure systématiquement isolé ou sous-pondéré et, le cas échéant, forcer une mise à jour corrective. De tels audits peuvent se fonder sur l'inspection des pondérations  $\{\omega_{i,j}\}$  associées à un ensemble d'entités. Ils peuvent également imposer une mesure d'équité en tant que contrainte externe, qui intervient de façon hiérarchique pour éviter la constitution de partitions inéquitables. L'objectif demeure le **contrôle** du **DSL** tout en respectant l'esprit d'**auto-organisation**, afin de combiner l'efficacité de la synergie et la nécessité de préserver une **égalité** ou une **justice** dans les décisions rendues.

#### 1.7.6.3. Protection des Données et Droit à l'Oubli

Dans une architecture **Deep Synergy Learning (DSL)**, il arrive fréquemment que le réseau comprenne des **entités** liées à des personnes (utilisateurs, patients). Les législations comme le **RGPD** (Règlement Général sur la Protection des Données) en Europe imposent plusieurs obligations, dont la **minimisation** des données recueillies (ne conserver que l'essentiel), le **droit à l'oubli** (un individu peut exiger la suppression de ses données personnelles) et des règles d'**anonymisation** ou de pseudonymisation pour éviter la ré-identification. Le réseau DSL, en tant que graphe auto-organisé et évolutif, doit donc intégrer des procédures garantissant que l'on puisse effacer ou rendre inopérante l'entité  $\mathcal{E}_i$  correspondant à une personne qui souhaite être oubliée.

La présence d'une entité  $\mathcal{E}_i$  dans un **Synergistic Connection Network** peut avoir des répercussions à de multiples endroits, car l'**auto-organisation** a pu former plusieurs **clusters** où  $\mathcal{E}_i$  joue un rôle. Pour satisfaire le droit à l'oubli, il faut d'abord **retracer** cette entité dans le graphe, puis la **retirer** du réseau ou la rendre totalement anonyme. Il convient de supprimer ou de neutraliser les liaisons  $\omega_{i,j}$  reliées à l'entité  $\mathcal{E}_i$ . Lorsque l'on procède à cette suppression, on doit parfois **réajuster** la **synergie** locale afin que les entités  $\mathcal{E}_j$  anciennement liées à  $\mathcal{E}_i$  ne se retrouvent pas avec des pondérations invalides. L'opération de "déréférencement" peut donc influencer sur l'équation d'évolution, qui doit détecter et ignorer toute trace associée à  $\mathcal{E}_i$ . L'objectif est d'éviter que des **résidus** dans la structure du réseau ne permettent de re-identifier l'utilisateur a posteriori, contrevenant ainsi aux principes de confidentialité exigés par la loi.

Le RGPD recommande également la **pseudonymisation** et la **minimisation** des informations stockées, ce qui affecte la façon dont les entités DSL sont construites. Il est parfois nécessaire de ne stocker, dans une entité  $\mathcal{E}_i$ , qu'une version agrégée (par exemple, localisation au niveau d'une ville plutôt que d'une rue) et d'exclure tout identifiant direct pour se conformer aux dispositions légales. Si le réseau DSL garde un trop haut niveau de précision, il peut être considéré comme non conforme à la réglementation. Une possibilité consiste à mettre en place un **processus** d'anonymisation périodique : après un certain temps, on fusionne des entités trop granulaires ou on supprime les

éléments de détail, en sacrifiant une partie de la granularité tout en conservant l'essence des **synergies** globales. Cette pratique garantit une protection de la vie privée tout en permettant au DSL de continuer à fonctionner de manière auto-organisée et réactive, mais sur la base de données moins sensibles ou déjà anonymisées.

#### 1.7.6.4. Transparence, Explicabilité et Contrôle

Dans certains contextes réglementés, comme le **RGPD** ou des lois spécifiques à un domaine (médical, bancaire, etc.), les usagers ou patients ont la faculté d'exiger une **explication** sur la décision qui les concerne. Le **Deep Synergy Learning (DSL)**, grâce à ses **clusters** et à ses **liaisons** plus faciles à interpréter qu'un pipeline neuronal opaque, peut répondre à cette exigence. Il demeure toutefois nécessaire de s'assurer que l'évolution du réseau (les pondérations  $\omega_{i,j}$  dans le Synergistic Connection Network) demeure **traçable** dans le temps. Pour cela, on peut tenir un **historique** des principales mises à jour ou conserver des **snapshots** à intervalles réguliers, afin de justifier a posteriori comment tel ou tel sous-réseau s'est formé.

Une **infrastructure** d'explication (voir section 1.7.5) doit de surcroît synthétiser le résultat de la **dynamique** auto-organisée. Il ne s'agit pas nécessairement d'exposer l'entièreté des liens  $\omega_{i,j}$ , mais d'extraire des **chemins** ou des **clusters** pivot permettant de reconstituer le **raisonnement local**. Une difficulté survient lorsque le système évolue rapidement, voire "chaotiquement" : dans ce cas, l'explication risque de changer sensiblement d'une itération à l'autre, ce qui nuit à la cohérence d'ensemble. Des **mécanismes** de stabilisation ou de **pas** d'évolution limités (contrôle de  $\eta$  et de  $\tau$ , par exemple) peuvent aider à préserver une relative continuité, rendant les explications plus consistantes dans le temps.

Pour de nombreuses **applications** sensibles, la législation ou la pratique courante impose un **contrôle humain** dans la boucle. Conformément à un principe d'audit ou de validation, un expert (médecin, analyste, manager) examine la configuration actuelle du DSL et doit pouvoir **accepter**, **refuser** ou **rectifier** certaines décisions. L'**interface** qui présente les clusters, leurs pondérations dominantes et l'historique succinct de leur formation doit donc être la plus claire possible, en hiérarchisant l'information. L'expert peut alors intervenir en imposant une pénalisation sur des liaisons considérées comme non conformes (sécurité, éthique, cohérence métier). Cette démarche articule la liberté **auto-organisée** du DSL (qui adapte continuellement ses liaisons) avec un **pilotage** humain, assurant un respect des **cadres légaux** et éthiques, tout en tirant parti de la plasticité et de l'auto-organisation qu'offre le Deep Synergy Learning.

#### Conclusion

Les **considérations éthiques et réglementaires** forment un **pilier** incontournable lors du déploiement de **systèmes auto-organisés** comme le DSL. On peut résumer les **grandes lignes** de vigilance :

67. **Responsabilité et traçabilité** : éviter qu'un système auto-adaptatif devienne totalement opaque quant à ses décisions ou conséquences, mettre en place un journal d'événements, un module de supervision.
68. **Biais et équité** : surveiller la formation de clusters discriminants, adapter la dynamique pour prévenir les exclusions ou dominations injustes.
69. **Protection des données** : garantir l'anonymisation, la pseudonymisation, et le droit à l'oubli dans un graphe évolutif (ce qui n'est pas trivial techniquement).
70. **Transparence et droit à l'explication** : mettre en place des mécanismes d'explicabilité (voir 1.7.5) assurant que l'utilisateur comprend la logique du DSL.

Au final, l'**originalité** du DSL ne saurait dispenser d'une **responsabilisation** solide. La recherche sur l'IA dite "responsable" ou "de confiance" s'applique tout autant (voire davantage) à un modèle **auto-organisé** qui, par définition, échappe aux schémas de contrôle univoque. Les prochaines sections (et développements futurs) devront approfondir les outils et protocoles pour faire du DSL une **technologie** à la fois puissante, adaptable, et conforme aux **valeurs** et **règles** en vigueur.

#### 1.7.7. Comparaisons Expérimentales avec d'Autres Approches

Dans les sections précédentes (1.7.1 à 1.7.6), nous avons exploré les **défis** et **contraintes** du **Deep Synergy Learning (DSL)** en matière de **complexité**, de **qualité des données**, de **stabilité**, d'**explicabilité**, et d'**éthique**. Pour affermir la **pertinence** de ce paradigme sur le terrain, il est indispensable d'effectuer des **comparaisons expérimentales** rigoureuses avec des méthodes existantes : réseaux neuronaux profonds (CNN, RNN, Transformers), méthodes de clustering (k-means, DBSCAN), algorithmes d'optimisation (ordonnancement, etc.), voire des approches neuro-symboliques plus classiques. Cette sous-section (1.7.7) aborde :

71. Les **critères** de comparaison essentiels (performances quantitatives, robustesse, adaptativité...),

72. Les **protocoles** expérimentaux pour confronter le DSL à d'autres techniques (tests sur données stationnaires et non stationnaires, analyses de complexité, etc.),
73. Les **scénarios** où le DSL a un net avantage (auto-organisation, peu de supervision, scénarios évolutifs),
74. Les **limites** et enseignements que de tels benchmarks peuvent révéler, guidant l'amélioration du DSL.

#### 1.7.7.1. Choisir les Critères de Comparaison

Dans la plupart des tâches d'apprentissage ou de décision (vision, audio, recommandation, diagnostic médical), on évalue la **précision** ou des métriques dérivées comme l'**accuracy**, la **F1-score** ou l'**AUC**. Lorsque l'on souhaite comparer un **Deep Synergy Learning (DSL)** avec, par exemple, un CNN ou un Transformer, on peut se borner à mesurer son **taux de reconnaissance** (dans un scénario de classification d'images) ou la **sensibilité** et la **spécificité** (dans le cadre d'un diagnostic médical). Toutefois, ces **scores** globaux ne reflètent pas nécessairement la **richesse** de la structure auto-organisée du DSL, qui excelle par sa **plasticité** et son adaptation continue, sans exiger nécessairement de labels massifs. Il est donc essentiel, lors de l'évaluation, d'incorporer des critères tenant compte de la **dynamique** de l'auto-organisation.

Le **DSL** se veut **adaptatif** en présence de bruit, de pannes de capteurs ou de changements de distribution (concept drift). Pour mettre en évidence cette capacité, il est concevable de concevoir des expériences où la distribution des données varie progressivement ou subitement en cours d'apprentissage. Dans un cas classique, un **réseau neuronal** (CNN, RNN) va souvent exiger une **séance de re-fine-tuning** global, tandis que le DSL doit pouvoir **réajuster** localement ses liaisons  $\{\omega_{i,j}\}$ . On peut alors comparer la **vitesse de réadaptation** et la **qualité** de la reprise de performance. On mesure, par exemple, la **chute** temporaire de précision lorsqu'un nouveau type de données apparaît, ainsi que le **temps** qu'il faut au modèle pour retrouver un niveau satisfaisant. Le DSL a pour atout de pouvoir réorganiser ses clusters et de créer ou rompre des liens en fonction des synergies locales, ce qui peut se révéler plus rapide et moins coûteux qu'un re-entraînement complet.

La **taille** du modèle (nombre de paramètres ou de liaisons actives) et la **charge** de calcul (temps CPU, mémoire GPU) comptent également parmi les indicateurs déterminants. Un DSL peut rester **parcimonieux** s'il opère une **sparsification** rigoureuse de la matrice  $\omega$ . Si, au contraire, on ne limite pas la prolifération des liens, la structure risque de gonfler et de devenir onéreuse en ressources. L'évaluation doit donc englober la **scalabilité** et l'**efficacité** : on vérifie si, pour un grand nombre d'entités, le DSL parvient à maintenir une densité de liens gérable (voir section 1.7.1) et à répondre aux scénarios évolutifs (pannes, modifications de distribution) avec des coûts de mise à jour et de mémoire acceptables.

#### 1.7.7.2. Protocoles Expérimentaux Envisageables

Pour **évaluer** la pertinence d'un **Deep Synergy Learning (DSL)** dans des scénarios variés, on peut le confronter à des bases de données classiques ou à des cas pratiques bien établis :

- Des ensembles **d'images** (MNIST, CIFAR, éventuellement un sous-ensemble d'ImageNet) pour évaluer la classification ou la détection, en analysant la **précision** obtenue par l'auto-organisation visuelle des entités (patches, features, etc.) face à un CNN ou un Transformer.
- Des ensembles **audio** (LibriSpeech, UrbanSound8K) pour tester la reconnaissance de sons ou la transcription de parole, où le DSL formerait des entités "segment audio" et "caractéristiques acoustiques", comparées à un système RNN ou CNN entraîné de façon traditionnelle.
- Des **bases de recommandation** (MovieLens, Amazon reviews) pour mesurer la qualité des suggestions formulées par le DSL ; on confronte alors la structure auto-organisée (clusters d'utilisateurs, de contenus, etc.) aux méthodes de factorisation matricielle ou aux modèles de collaboration neuronaux, en examinant le **Recall@K**, la **Precision@K** ou la **NDCG**.
- Des **données médicales** (ensembles de pathologies, EHR) afin de juger la capacité du DSL à découvrir des clusters pathologiques ou à émettre des hypothèses de diagnostic, comparativement à un modèle supervisé (réseau profond, arbre d'ensemble) ou semi-supervisé.
- Des **scénarios robotiques** (simulateurs comme Gazebo, PyBullet, ou environnement multi-agents) pour vérifier la coordination sensorimotrice : un DSL peut relier capteurs et effecteurs et montrer sa robustesse dans la détection de pannes ou l'évolution de conditions.

Dans chacun de ces domaines, on confronte les **résultats** finaux (taux de reconnaissance, score de recommandation, F1-score diagnostique, etc.) à ceux de modèles **classiques** : CNN, RNN, Transformer, clustering K-means, factorisation matricielle, etc. On veille aussi à mesurer la **taille** du modèle (paramètres, liaisons), la **complexité** de calcul et la **vitesse d'inférence**.

Le **DSL** se veut également **adaptatif** en cas de changements de distribution (concept drift), de pannes de capteur, de conditions altérées. Pour l'exercer à cette fin, il est utile de mettre en œuvre des **scénarios** où les données subissent un glissement graduel ou une modification brutale (apparition de nouvelles classes, hausse du bruit, changement d'éclairage, etc.). Sur la timeline  $\{t_0, \dots, t_f\}$ , on évalue :

75. La manière dont un réseau neuronal classique (CNN, RNN) gère ce drift (souvent par un re-fine-tuning ou un forget partiel).

76. La façon dont le DSL, sans réapprentissage massif, modifie localement ses liaisons  $\omega_{i,j}(t)$ .

On observe la **performance** en continu (taux de classification, AUC, indice de recommandation) et on étudie la **vitesse** de réadaptation. Le **DSL** devrait, en principe, reporter des ajustements plus localisés et moins coûteux qu'un re-entraînement total, tout en préservant une cohérence globale. Ceci permet de vérifier empiriquement que la **robustesse** et la **plasticité** conférées par la mise à jour distribuée des pondérations  $\omega$  confèrent un avantage aux méthodes d'auto-organisation, surtout dans des domaines où les données ou le contexte varient régulièrement.

### 1.7.7.3. Scénarios où le DSL a un Avantage Notable

Le **Deep Synergy Learning (DSL)** se distingue particulièrement dans des environnements où l'on dispose de **peu ou pas de labels** (section 1.5.5) ou bien où la supervision est coûteuse et incomplète. En mode **non supervisé** ou faiblement supervisé, un réseau neuronal classique (tel qu'un CNN) peine à extraire des structures cohérentes, car il manque d'exemples étiquetés pour guider l'apprentissage. Le **DSL**, au contraire, s'appuie sur sa **fonction de synergie**  $S(\mathcal{E}_i, \mathcal{E}_j)$  pour **auto-organiser** les entités, sans nécessiter un volumineux jeu de labels. Les **clusters** émergent alors de la simple coopération (ou co-occurrence) observée entre entités, ce qui rend le réseau particulièrement apte aux tâches d'**exploration** et de **regroupement** de données hétérogènes.

Un autre atout du DSL s'observe en **multi-modalité hétérogène** (section 1.5.2), où l'on combine image, audio, texte, voire des flux sensoriels divers. Les approches traditionnelles imposent souvent un **pipeline** de fusion rigide (concatenation tardive, attention inter-modale) dépendant de choix d'architecture prédéfinis. Le DSL adopte une démarche plus flexible : chaque entité  $\mathcal{E}_i$  (visuelle, auditive, etc.) forme des **liaisons**  $\omega_{i,j}$  spontanées selon la synergie perçue. Cette structure évite la planification d'un module de "fusion" spécifique ; la mise en réseau est régie par la **dynamique** locale des pondérations, autorisant des clusters multimodaux à émerger naturellement.

Lorsque les **flux** de données sont **dynamiques** et sujets à des **pannes** ou à des transformations (sections 1.5.4 et 1.7.2.3), le DSL illustre également sa supériorité. Un capteur défectueux voit ses liaisons  $\omega$  chuter, car la synergie avec le reste du système devient faible. Le réseau se réorganise alors sans qu'un nouvel entraînement global ne soit requis, contrairement à un réseau neuronal classique où une modification durable de la distribution (panne de capteur, concept drift) nécessite un **fine-tuning** ou un recalibrage important.

L'**explicabilité** du DSL revêt aussi un intérêt clé (sections 1.7.5 et 1.5.6). Les **clusters** ou les chemins de **synergie** identifiés entre entités procurent des explications plus détaillées qu'une simple "heatmap" d'attention. Dans un réseau traditionnel, la structure interne (couches, neurones) reste souvent opaque, malgré certaines techniques de visualisation de filtres. Le DSL, en attribuant des **identités** distinctes à ses nœuds et en laissant les liens  $\omega_{i,j}$  témoigner de leur degré de coopération, permet une **compréhension** plus directe de la raison pour laquelle tel sous-réseau est né, ou tel couple  $(\mathcal{E}_i, \mathcal{E}_j)$  a vu sa liaison renforcée.

Dans l'ensemble, une **comparaison expérimentale** (section 1.7.7.2) révélera que, face à un CNN ou à un Transformer, le DSL excelle dans des conditions de **data faible** ou de **distribution changeante**, ou encore quand l'on souhaite un **rendement** sémantique mieux explicité par un graphe de clusters. Ces propriétés séduisent dans des scénarios où l'on doit adapter le système en continu, respecter des contraintes d'équité, ou justifier les décisions auprès d'utilisateurs ou d'experts.

### 1.7.7.4. Limites et Enseignements Possibles

Lors de l'évaluation d'un **Deep Synergy Learning (DSL)** à large échelle, on peut constater que la structure du réseau, qui autorise jusqu'à  $O(n^2)$  liaisons dans le cas binaire, devient difficile à manipuler si les entités sont très nombreuses. Il peut être nécessaire d'imposer une **sparsification** stricte (par exemple, ne maintenir qu'un voisinage restreint pour chaque entité) ou de recourir à un **algorithme local** qui ne calcule pas la synergie entre toutes les paires d'entités, mais seulement entre celles jugées proches dans un espace de représentation. Les expérimentations comparatives, que ce soit en temps de calcul ou en mémoire GPU, permettent d'identifier le **seuil** au-delà duquel le DSL se montre moins performant que d'autres architectures classiques (CNN, RNN, Transformer) optimisées pour le passage à l'échelle sur des supports matériels massivement parallèles.

Par ailleurs, si l'on dispose d'un **jeu de données étiquetées** gigantesque (par exemple la totalité d'ImageNet, avec plus de 14 millions d'images), un réseau supervisé standard tel qu'un CNN ou un Transformer minutieusement entraîné peut atteindre d'excellentes performances de classification top-n. Le **DSL**, plus orienté vers l'**auto-organisation** non supervisée ou faiblement supervisée, pourrait dans ce cas demeurer en retrait si l'on n'exploite pas pleinement toute la supervision à disposition. On peut envisager des versions supervisées ou semi-supervisées du DSL où la synergie est partiellement guidée par des labels, mais la question demeure ouverte : ces labels suffiront-ils à faire concurrence à un CNN entraîné par descente de gradient pure, surtout lorsque le volume d'annotations est massif ? Ces expériences permettront de clarifier dans quelle mesure le DSL parvient à tirer parti d'une supervision large sans renoncer à sa plasticité et à sa modularité intrinsèque.

Ces limites et enseignements soulignent la nécessité de concevoir des **protocoles hybrides**, dans lesquels on combine la liberté d'auto-organisation du DSL (favorable à l'adaptation continue ou à la gestion de scénarios peu labellisés) avec la force d'un apprentissage

supervisé classique lorsqu'un dataset étiqueté d'ampleur est disponible. C'est dans cet espace de compromis que réside probablement l'avenir des approches synergiques à l'échelle industrielle.

## Conclusion

La **validation expérimentale** du **DSL** face aux approches dominantes (réseaux neuronaux profonds, clustering, méthodes d'optimisation) est incontournable pour :

- **Quantifier** la performance, la robustesse, la capacité d'adaptation,
- **Identifier** les atouts (apprentissage sans label massif, gestion du drift, multimodalité hétérogène),
- **Révéler** les limites (coût en calcul, besoin de mesures de synergie fiables, difficulté sur des tâches supervisées massives).

Des **benchmarks** variés (vision, audio, recommandation, diagnostic, robotique) permettront de **cartographier** où le DSL s'impose et où il doit s'allier à d'autres techniques (CNN pré-entraîné pour extraire des features, par exemple). Au final, ce travail de confrontation expérimentale, mené en parallèle des réflexions éthiques (1.7.6) et de la recherche sur l'optimisation (1.7.3), consolide la position du DSL comme un **paradigme** original — potentiellement complémentaire, voire compétitif — dans le paysage actuel de l'apprentissage automatique.