

Titre de la thèse (sur plusieurs lignes si nécessaire)

*Traduction du titre de la thèse (sur plusieurs lignes si
nécessaire)*

**Thèse de doctorat de l'université Paris-Saclay et de l'université XXX (si
cotutelle - sinon enlever cette seconde partie)**

École doctorale n° d'accréditation, dénomination et sigle

Spécialité de doctorat : voir annexe

Graduate School : voir annexe. Référent : voir annexe

Thèse préparée dans la (ou les) unité(s) de recherche **Nom(s)** (voir annexe), sous la
direction de **Prénom NOM**, titre du directeur ou de la directrice de thèse, la co-direction
de **Prénom NOM**, titre du co-directeur ou de la co-directrice de thèse, le co-encadrement
de **Prénom NOM**, titre, du co-encadrant ou de la co-encadrante ou la co-supervision de
Prénom NOM, titre, du tuteur ou de la tutrice (en cas de partenariat industriel)

Thèse soutenue à Paris-Saclay, le JJ mois AAAA, par

Prénom NOM

Composition du jury

Membres du jury avec voix délibérative

Prénom NOM
Titre, Affiliation
Prénom NOM
Titre, Affiliation
Prénom NOM
Titre, Affiliation
Prénom NOM
Titre, Affiliation
Prénom NOM
Titre, Affiliation

Président ou Présidente
Rapporteur & Examineur / trice
Rapporteur & Examineur / trice
Examineur ou Examinatrice
Examineur ou Examinatrice

Titre : titre (en français).....

Mots clés : 3 à 6 mots clefs (version en français)

Résumé : Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci

eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Title : titre (en anglais).....

Keywords : 3 à 6 mots clefs (version en anglais)

Abstract : Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci

eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Table des matières

1	Introduction	7
2	Problématique (plan)	9
3	Problématique	11
3.1	Introduction	11
3.2	Description de l'intégration du capteur de Mesures de Soutien Électronique et son fonctionnement	11
3.2.1	Contexte opérationnel : La maîtrise du spectre électromagnétique	11
3.2.2	Chaîne de traitement de l'information	11
3.2.3	Architecture et fonctionnement du capteur	12
3.3	Description de l'environnement numérique	13
3.3.1	Motivations	13
3.3.2	Architecture et fonctionnement de l'environnement numérique	14
3.4	Identification du goulot d'étranglement et limites opérationnelles	15
3.4.1	L'impératif de temps réel et le coût de la simulation	15
3.4.2	Analyse de la complexité et localisation du verrou	16
3.5	Formalisation et complexité du problème d'apprentissage	16
3.5.1	Nature des données et définition formelle de la tâche	16
3.5.2	Une double problématique : Traitement de séquence et Génération	17
4	État de l'art : PLAN (à supprimer après rédaction)	19
4.1	Introduction	19
4.2	IA générative	19
4.3	Méthodes pour le traitement de séquence	19
4.4	Les améliorations	20
5	État de l'art	21
5.1	Introduction : Environnements Numériques et typologie des apports de l'IA	21
5.1.1	Cadre Conceptuel : Environnement Virtuel et Jumeau Numérique	21
5.1.2	L'IA pour la constitution géométrique et visuelle de l'environnement	23
5.1.3	L'IA pour l'accélération et la modélisation des phénomènes physiques	24
5.1.4	L'IA au service de l'interactivité et de l'adaptation décisionnelle	25
5.1.5	Ancrage dans la problématique	26
5.2	IA générative	26
5.2.1	L'approche probabiliste explicite : Les VAE (2013)	27
5.2.2	La révolution antagoniste : Les GAN (2014)	27
5.2.3	La génération par raffinement : Les Modèles de Diffusion (2020)	28
5.2.4	Le paradigme séquentiel et l'Autorégression	28

5.2.5	Ancrage dans la problématique	29
5.3	Méthodes de traitement : Séquences et structures spatiales	29
5.3.1	Typologie des données : De la causalité temporelle à la topologie spatiale	29
5.3.2	Réseaux de convolution	30
5.3.3	Réseaux de neurones récurrents et Espaces d'Etats (RNN et SSM)	35
5.3.4	Transformer	39
5.3.5	L'architecture Transformer	47
5.3.6	Ancrage dans la problématique	54
6	AVERTISSEMENT	57
7	COMPOSITION GÉNÉRALE, CHARTE GRAPHIQUE	59
7.1	COMPOSITION DU DOCUMENT	59
7.2	QUELS LOGOS FAIRE FIGURER?	59
7.3	POLICES DE CARACTÈRES ET COULEURS	59
8	INFORMATIONS GÉNÉRALES SUR LA PAGE DE COUVERTURE	61
8.1	TITRE DE LA THÈSE ET LANGUE(S)	61
8.2	SPÉCIALITÉ DE DOCTORAT	61
8.3	UNITÉ DE RECHERCHE	62
8.4	LE RÉFÉRENT	62
8.5	GRADUATE SCHOOL	63
8.6	ÉCOLE DOCTORALE	63
8.7	LIEU ET DATE DE SOUTENANCE	64
9	CIVILITÉ, FÉMINISATION DES TITRES ET FONCTIONS	65
10	PRÉSENTATION DE LA DIRECTION DE LA THÈSE OU DE L'ÉQUIPE D'ENCADREMENT	67
11	COMPOSITION DU JURY	69
11.1	A QUOI SERVENT CES INFORMATIONS?	69
11.2	LÉGITIMITÉ ACADÉMIQUE	69
11.3	INDÉPENDANCE	70
11.4	FONCTION DANS LE JURY ET ORDRE DE CITATION	71
11.4.1	Ordre de citation	71
11.4.2	Les rapporteurs	71
12	BIEN CITER SES SOURCES	73
12.1	S'INFORMER SUR LE PLAGIAT	73
12.2	LES IMAGES	73
12.3	ARTICLES JOINTS A LA THÈSE	73
13	DÉPOSER ET DIFFUSER SA THÈSE	75
13.1	LES RESSOURCES A CONSULTER	75
13.2	LES DÉMARCHES	75

14 ANNEXE : LES LOGOS INSTITUTIONNELS	77
14.1 LE LOGO DE L'UNIVERSITÉ PARIS-SACLAY	77
14.2 LOGOS, NUMÉROS D'ACCRÉDITATION ET DÉNOMINATIONS DES ÉCOLES DOCTORALES	77

1 - Introduction

Le chapitre introduction comprendra les éléments suivants :

- Introduction du domaine de la guerre électronique
- Les grandes lignes du rôle du capteur ESM
- Explications de l'utilité de l'environnement numérique
- La problématique d'accélération
- Introduction succincte sur le domaine de l'IA
- Le plan de notre approche

2 - Problématique (plan)

Le chapitre sur la problématique contiendra les éléments suivants :

- Description du fonctionnement du capteur ESM dans l'environnement (dans la limite de ce qu'on peut dire), intégré dans la chaîne algorithmique de traitement de l'information.
- Description du fonctionnement de l'environnement numérique, avec l'explication des modélisations de chaque traitement.
- Spécification du goulot d'étranglement et commentaire sur les données I/O.
- Commentaire sur la complexité du problème pour l'apprentissage automatique : double problématique génération et traitement de séquence.
- Penser à ajouter des références de traitement avec ce principe de mesureur (brevet?) pour montrer qu'on ne révèle pas des secrets

3 - Problématique

3.1 . Introduction

Comme abordé dans le chapitre introductif, notre mission est d'accélérer la simulation d'un environnement numérique modélisant l'interception d'impulsion RADAR par des capteurs ESM. Ce chapitre va nous permettre de revenir sur cette problématique. Nous commencerons par l'explication du fonctionnement du capteur ESM et son intégration dans le chaîne de traitement de l'information. Nous verrons ensuite pourquoi il est nécessaire de disposer d'un environnement numérique modélisant le fonctionnement du capteur et nous reviendrons sur cette modélisation. Après nous identifierons le goulot d'étranglement et précierons alors la nature exacte du problème d'accélération que nous aurons à résoudre. Nous porterons une attention particulière à la lecture de se problème sous le spectre de l'apprentissage automatique en notant que l'aspect est à la fois traitement de séquence et génération. Finalement, nous exposerons les références des problèmes où ce type de traitement apparaît, des brevets existants, etc

3.2 . Description de l'intégration du capteur de Mesures de Soutien Électronique et son fonctionnement

3.2.1 . Contexte opérationnel : La maîtrise du spectre électromagnétique

Dans le cadre d'un scénario de guerre électronique, la survie de l'aéronef dépend de sa capacité à percevoir et comprendre son environnement électromagnétique. L'appareil évolue dans un espace abondant d'émissions provenant de RADAR adverses ou civils, au sol ou aéroportés, cherchant eux-mêmes à détecter leur cible. Les caractéristiques techniques de ces signaux, telles que la fréquence, la largeur d'impulsion ou la période de répétition, constituent une signature unique permettant d'identifier l'émetteur et d'en déduire ses intentions tactiques (veille, poursuite, engagement). C'est la mission des capteurs de Mesures de Soutien Électronique (ESM - Electronic Support Measures) : assurer une écoute passive et discrète du spectre pour détecter, caractériser et localiser ces menaces potentielles, fournissant ainsi les données critiques à la décision stratégique et aux contre-mesures.

3.2.2 . Chaîne de traitement de l'information

L'intégration du capteur s'inscrit dans une architecture de traitement séquentielle visant à transformer un signal physique brut en renseignement tactique exploitable. Le champ électromagnétique incident est initialement capté par le capteur ESM, qui opère la première conversion fondamentale : il détecte les impulsions radar et construit en temps réel des descripteurs numériques, les PDW (Pulse Description Word - Mot de Description d'Impulsion). Chaque PDW synthétise les paramètres mesurés de l'impulsion : Date d'arrivée (ToA), Largeur d'impulsion (LI), Fréquence, Niveau de puissance et Direction d'arrivée (DoA).

Ce flux continu de PDW alimente ensuite les algorithmes de traitement de haut niveau. Une étape de désentrelacement regroupe d'abord les impulsions par émetteur sur un horizon temporel court, isolant les trains d'impulsions cohérents. Ces regroupements élémentaires sont ensuite consolidés par un processus de pistage (tracking) qui suit l'évolution des émetteurs sur le long terme pour en caractériser la cinématique et le mode de fonctionnement. Finalement, ces pistes enrichies sont confrontées à des bases de données de signatures pour identifier formellement le système d'arme associé et évaluer le niveau de menace immédiat.

3.2.3 . Architecture et fonctionnement du capteur

Le fonctionnement interne du capteur ESM ne se limite pas à une conversion analogique-numérique transparente; il constitue une chaîne complexe de traitements physiques et logiques qui conditionne structurellement la qualité des données produites. Les traitements décrits ci-après constituent le socle architectural de la plupart des récepteurs numériques modernes, bien que des variantes d'implémentation, dictées par les contraintes matérielles des systèmes temps réel, puissent exister selon les constructeurs.

Le traitement débute par la conversion du champ électromagnétique incident en signal électrique analogique. L'objectif est ensuite d'extraire, sur des fenêtres temporelles successives, les raies spectrales significatives. La mission de surveillance de bandes passantes instantanées de plusieurs gigahertz (typiquement $2 - 18GHz$) impose l'usage de bancs de convertisseurs analogique-numérique (au moins 3) fonctionnant en parallèle à des cadences inférieures à la fréquence de Nyquist (ex : $1GHz$, $1.2GHz$, $1.4GHz$) [1]. Ce choix architectural induit un repliement spectral systématique sur chaque voie d'acquisition, mais permet la détermination de la fréquence réelle grâce à la résolution d'un système de congruences entre les différentes voies repliées, principe connu sous le nom de théorème des restes chinois [2], [3]. Cependant, sur chaque canal d'acquisition, des phénomènes de masquage peuvent intervenir, liés à la saturation des convertisseurs, aux harmoniques et/ou à l'intermodulation provoquée par des impulsions de haute énergie, ou à la proximité fréquentielle entre signaux repliés. De plus, la résolution d'ambiguïté s'appuie généralement sur une sélection restreinte des N pics spectraux les plus énergétiques par canal, liée aux contraintes matérielles. La conjonction du bruit thermique sur des canaux critiques et de cette sélection limitative peut conduire à l'échec de la levée d'ambiguïté, entraînant la perte d'impulsions sur la fenêtre temporelle considérée. Les fréquences non-ambiguës identifiées, associées leur énergie, sont qualifiées de détections non-ambiguës (DNA).

Une fois les DNA identifiées sur la fenêtre d'analyse courante, elles sont transmises collectivement au système de suivi temporel. Ce module reçoit ainsi un paquet de détections simultanées qu'il doit associer aux ressources mémoires actives, qualifiées de "pistes" [4]. Le processus d'association est séquentiel et hiérarchisé : les DNA sont traitées une à une, par ordre de priorité croissante selon leur amplitude, afin de privilégier le suivi des signaux les plus énergétiques. Pour chaque détection candidate, le système recherche une correspondance parmi les pistes actives sur la base de critères de tolérance prédéfinis (proximité fréquentielle, cohérence de niveau). Une contrainte stricte d'unicité s'applique alors : une piste ne peut être mise à jour qu'une seule fois par fenêtre temporelle. En cas de corrélation valide avec une piste disponible (non

encore mise à jour sur ce cycle), celle-ci intègre les nouveaux paramètres et prolonge sa durée de vie. Si aucune association n'est possible, ou si la piste candidate a déjà été servie par une détection prioritaire, une nouvelle piste est ouverte pour le signal, sous réserve qu'une ressource mémoire soit libre. La clôture d'une piste et la génération du PDW final [5] s'opèrent finalement lorsqu'elle n'a pas reçu de mise à jour durant une période seuil ou lorsque l'impulsion dépasse sa limite de durée opérationnelle.

La limitation matérielle du nombre de ces mémoires, conjuguée à leur logique d'allocation, engendre des artefacts de segmentation spécifiques. Premièrement, une contrainte de latence maximale impose de segmenter artificiellement les impulsions très longues ou continues pour assurer des mises à jour périodiques, générant une série de PDW contigus. Deuxièmement, les échecs de résolution d'ambiguïté peuvent provoquer une rupture de suivi prématurée. Si le masquage est transitoire, la piste reprend après une interruption, scindant l'impulsion en plusieurs entités. Si le masquage persiste jusqu'à la fin de l'émission, le suivi s'arrête définitivement, tronquant la fin du signal. Troisièmement, la saturation des ressources mémoires en environnement dense impacte directement la détection : si aucune piste n'est disponible à l'apparition du signal, il sera ignoré sur l'instant. Cela conduit soit à une acquisition tardive dès la libération d'une ressource, amputant alors le début du signal, soit à une perte totale de l'impulsion si aucune ressource ne se libère à temps. À l'inverse de ces phénomènes de fragmentation ou de perte, la résolution temporelle finie des bancs de filtres peut conduire à l'amalgame de deux impulsions brèves et rapprochées en un seul descripteur.

Ainsi, la séquence de PDW produite ne doit pas être considérée comme une simple mesure dégradée de la réalité, mais comme une reconstruction interprétée, portant intrinsèquement la signature des limitations fréquentielles et des heuristiques de gestion de ressources du capteur.

3.3 . Description de l'environnement numérique

3.3.1 . Motivations

La phase d'Intégration, Vérification, Validation et Qualification (IVVQ) des algorithmes de guerre électronique, tels que le désentrelacement, le pistage et l'identification, exige de disposer de jeux de données rigoureusement contrôlés. Pour valider la chaîne de traitement, il est nécessaire de confronter les données d'entrée des algorithmes - le flux de PDW en sortie du capteur ESM - aux données de sortie attendues, c'est-à-dire la situation tactique restituée. Or, l'obtention de ces données par des essais en vol réels se heurte à des contraintes majeures. En effet, la connaissance exacte et exhaustive de la situation tactique (la position et l'activité de tous les émetteurs environnants) est souvent impossible à garantir, empêchant d'établir une "vérité terrain" fiable pour qualifier les algorithmes. De plus, la réalisation d'essais en vol dédiés représente un défi logistique et financier considérable. La constitution d'un scénario réaliste implique le déploiement de moyens conséquents, tels que des avions plastrons ou des stations radars au sol, dont la disponibilité est limitée. Par ailleurs, ces essais physiques ne permettent de couvrir qu'un spectre restreint de configurations géométriques et électromagnétiques, limitant la diversité des données récoltées. Face à ces obstacles, le recours à un simulateur d'impulsions s'impose comme la solution de référence. En permettant de générer le flux exact d'impulsions

que les capteurs ESM auraient intercepté pour un scénario prédéfini, l'environnement numérique offre une maîtrise totale des paramètres d'entrée. Cette approche autorise la simulation d'une quantité massive de données et la construction de scénarios d'une complexité arbitraire, incluant des cas limites difficilement reproductibles en vol. Cette capacité de tests intensifs est indispensable pour analyser finement la réaction de la chaîne algorithmique et procéder aux itérations nécessaires à son amélioration.

3.3.2 . Architecture et fonctionnement de l'environnement numérique

L'environnement numérique est structuré en quatre modules fonctionnels séquentiels. Il repose sur une approche de modélisation comportementale, dont l'objectif n'est pas de reproduire le traitement du signal échantillon par échantillon, mais de simuler l'effet macroscopique des traitements physiques et logiques sur le flux d'impulsions incident, afin de prédire les PDW que le capteur aurait effectivement générés.

Le premier module assure la génération de la vérité terrain électromagnétique. Il ingère les fichiers de description cinématique (trajectoires du porteur et des radars environnants) ainsi que les séquences d'émission théoriques de chaque radar. À partir de ces données, il construit une liste chronologique d'impulsions émises, triées par date d'arrivée (ToA), où chaque impulsion est décrite par un PDW idéal contenant ses caractéristiques physiques natives. Le second module modélise la chaîne de propagation et de réception analogique. Sa fonction est double : premièrement, il applique l'équation du bilan de liaison pour déterminer l'amplitude du signal arrivant au voisinage du porteur, en tenant compte de la position relative émetteur-récepteur et de la direction d'émission. Deuxièmement, il simule la fonction de transfert des antennes réceptrices. En exploitant les diagrammes de gain et l'angle d'incidence du signal, ce module calcule l'atténuation ou l'amplification subie par le signal électrique en sortie d'antenne. La sortie de ce bloc est une liste de PDW dont l'amplitude a été ajustée pour refléter la puissance réellement disponible à l'entrée du récepteur numérique.

Les deux derniers modules simulent le cœur du traitement numérique : la détection spectrale et la caractérisation temporelle. Pour s'abstraire de la simulation coûteuse du temps continu, ces modules opèrent sur une échelle temporelle discrétisée nommée "palier". Un palier définit un intervalle de temps au cours duquel l'environnement électromagnétique est considéré comme stationnaire : il est délimité par l'apparition ou la disparition d'une impulsion quelconque. Cette hypothèse de stationnarité permet de considérer que les fréquences détectables par le capteur restent constantes sur toute la durée du palier, autorisant un calcul unique par intervalle.

Le troisième module se charge de la modélisation spectrale à l'échelle de ce palier. Il reproduit les effets de l'architecture sous-Nyquist en calculant, pour chaque impulsion présente, ses fréquences repliées sur les différents canaux d'acquisition. En comparant les niveaux relatifs des signaux concurrents et en appliquant les seuils de sensibilité matériels, le module détermine la visibilité de chaque impulsion. Une impulsion est considérée comme "détectée" — c'est-à-dire que son ambiguïté fréquentielle aurait été levée avec succès — si elle reste visible

et non masquée sur au moins trois canaux simultanément, elle est alors qualifiée de DNA.

Enfin, le quatrième module reproduit le mécanisme de suivi temporel. Son fonctionnement mime la logique interne du capteur décrite précédemment, mais la cadence de mise à jour est ici dictée par les paliers et non par les fenêtres d'échantillonnage. À chaque palier, les DNA sont comparées aux pistes actives. En cas de corrélation, la piste est maintenue ; sinon, une nouvelle piste est allouée sous réserve de disponibilité mémoire. Ce module gère également les clôtures de pistes : si une piste n'est pas mise à jour pendant une durée critique, elle est fermée. De même, pour simuler la segmentation des signaux continus ou longs, une logique de coupure forcée est implémentée : lorsqu'une piste dépasse une durée maximale d'ouverture, elle est close (génération d'un PDW), puis immédiatement rouverte pour poursuivre le suivi, reproduisant ainsi fidèlement les artefacts de segmentation du capteur réel.

En conclusion, l'environnement numérique articule une chaîne de transformation cohérente qui mime le cycle de vie complet de l'information électromagnétique. En élaborant successivement une séquence de PDW idéaux issue de la vérité terrain, en leur appliquant les modulations radiométriques propres à la chaîne d'acquisition, puis en soumettant ce flux aux logiques de sélection spectrale et de suivi temporel du récepteur, le simulateur parvient à reproduire la séquence de PDW qu'un capteur aurait effectivement interceptée pour un scénario donné. Bien que cette modélisation comportementale constitue par essence une approximation par rapport à une simulation physique du signal au niveau de l'échantillon, elle offre un compromis optimal entre la précision des phénomènes reproduits et la charge de calcul. La représentativité des artefacts générés — incluant les effets de masquage, de fragmentation et de saturation — s'avère suffisante pour garantir la pertinence des données synthétiques, permettant ainsi de répondre aux exigences de diversité et de volume nécessaires à la validation robuste des algorithmes de traitement de l'information sans dépendre exclusivement des essais en vol.

3.4 . Identification du goulot d'étranglement et limites opérationnelles

3.4.1 . L'impératif de temps réel et le coût de la simulation

Si l'architecture modulaire de l'environnement numérique garantit une fidélité satisfaisante des données produites, son exploitation opérationnelle se heurte à une contrainte majeure de performance temporelle. Actuellement, la simulation détaillée des traitements du capteur présente un coefficient d'expansion temporel moyen de l'ordre de 100 : la simulation d'une seule seconde de scénario nécessite environ cent secondes de calcul. Cette latence prohibe l'utilisation du simulateur pour des applications critiques nécessitant une interaction en temps réel, telles que la formation des pilotes et des opérateurs de guerre électronique. Par ailleurs, même dans le cadre de la validation algorithmique hors ligne, ce coût calculatoire devient un obstacle dirimant pour la génération massive de données. La validation statistique robuste des algorithmes de désentrelacement ou d'identification exige de couvrir des milliers de variations de scénarios, une tâche qui, avec le coefficient d'expansion actuel, nécessiterait des temps de

calcul incompatibles avec les cycles de développement industriels, particulièrement pour les scénarios denses.

3.4.2 . Analyse de la complexité et localisation du verrou

L'analyse de profilage de l'environnement numérique permet de localiser précisément l'origine de cette latence au niveau des troisième et quatrième modules fonctionnels, responsables de la détection spectrale, du pistage et de la caractérisation des impulsions. Une distinction structurelle fondamentale sépare ces modules des étages amont. Les modules 1 et 2 (génération et propagation) appliquent des transformations physiques indépendantes sur chaque impulsion; ils se prêtent donc naturellement à une parallélisation sur CPU voire GPU. À l'inverse, les modules 3 et 4 opèrent intrinsèquement de manière séquentielle : l'état du système à l'instant t (les pistes actives, les masquages en cours) dépend de l'histoire du traitement, empêchant toute parallélisation temporelle simple. De surcroît, la complexité algorithmique de ces blocs est critique. Le nombre de paliers temporels à traiter croît linéairement avec la densité d'impulsions du scénario. Or, au sein de chaque palier, le module de détection doit effectuer des comparaisons croisées entre toutes les impulsions présentes pour résoudre les masquages et les ambiguïtés, induisant une complexité quadratique par rapport au nombre d'impulsions locales. Cette combinaison d'une structure séquentielle rigide et d'une complexité locale quadratique crée un goulot d'étranglement majeur dès que la densité du scénario augmente. Ce verrou, situé au cœur de la logique de traitement du capteur, constitue l'obstacle technologique qu'il est nécessaire de lever. L'objectif de la thèse est donc d'accélérer l'environnement numérique en substituant ces deux modules critiques, que nous désignerons conjointement comme le Sous-système de Traitement Numérique, par un modèle d'intelligence artificielle optimisé.

3.5 . Formalisation et complexité du problème d'apprentissage

3.5.1 . Nature des données et définition formelle de la tâche

L'objectif est de substituer au Sous-système de Traitement Numérique un modèle appris capable d'approximer sa fonction de transfert globale. Du point de vue des données, cette tâche se formalise comme un problème de transduction de séquence s'opérant dans un espace continu. L'entrée du modèle, notée X , est la séquence de PDW "modulés" issue du module de propagation (module 2). Elle représente l'information physique brute incidente aux bornes des convertisseurs analogique-numérique. La cible à prédire, notée Y , est la séquence de PDW "capteur" (sortie du module 4), correspondant aux impulsions effectivement construites et publiées par le système.

Ces données présentent deux caractéristiques structurelles fondamentales. Premièrement, elles appartiennent à un espace continu multidimensionnel. Chaque élément constitutif des séquences X et Y est un vecteur de valeurs réelles (Date, Fréquence, Largeur, Amplitude, Azimut) définissant les propriétés physiques de l'impulsion. Deuxièmement, les séquences se caractérisent par une cardinalité variable et asynchrone. La topologie temporelle diffère entre l'entrée

et la sortie en raison des mécanismes internes du capteur (masquages, fusions, scissions). Ainsi, la longueur N de la séquence d'entrée et la longueur M de la séquence de sortie sont variables et ne respectent pas de règle de proportionnalité stricte ou de synchronisation élément par élément.

3.5.2 . Une double problématique : Traitement de séquence et Génération

Cette caractérisation ancre le problème à l'intersection de deux paradigmes de l'apprentissage profond, induisant une complexité spécifique qui le distingue des applications classiques.

D'une part, il s'agit fondamentalement d'un problème de traitement de séquence (Sequence Processing), imposé par la nature variable et décorrélée des dimensions temporelles en jeu. Contrairement à des tâches de régression à taille fixe, le modèle doit traiter une série d'entrée de longueur arbitraire N pour produire une série de sortie de longueur M , sans qu'il existe de règle de proportionnalité simple entre N et M . Cette contrainte structurelle oblige l'architecture à s'abstraire de la dimension temporelle absolue pour raisonner sur des dépendances contextuelles relatives, nécessitant des mécanismes capables de mapper une séquence de taille variable vers une représentation latente fixe ou dynamique. De plus, la tâche se trouve complexifiée par la nature continue des données. Contrairement aux tâches de Traitement du Langage Naturel (NLP) où les séquences sont formées de mots discrets issus d'un dictionnaire fini, les PDW exigent une précision numérique fine sur plusieurs variables continues simultanément.

D'autre part, il s'agit d'un problème de génération conditionnelle. Le modèle ne doit pas simplement filtrer ou altérer les impulsions incidentes, mais construire intégralement une séquence de sortie dont la structure événementielle diffère fondamentalement de l'entrée. La logique de suivi temporel du capteur brise la relation bijective : un événement physique unique peut se trouver traduit par une succession de plusieurs PDW distincts, ou inversement, plusieurs événements peuvent être fusionnés. Le modèle doit donc apprendre à générer une nouvelle série de descripteurs qui constituent une interprétation synthétique de la réalité physique. Dans cette reconstruction, la séquence de sortie devient une entité structurellement autonome plutôt qu'une version simplement dégradée de l'entrée.

Cette dualité, combinée à l'exigence de précision, écarte les solutions sur étagère et nécessite la conception d'une architecture capable de marier les mécanismes d'attention globaux propres à la compréhension de contexte et les capacités de régression fine nécessaires à la reconstruction physique du signal.

4 - État de l'art : PLAN (à supprimer après rédaction)

Le chapitre sur l'état de l'art se découpe en 4 parties.

4.1 . Introduction

Cette section aborde les aspects suivants :

- Environnements numériques
- Injection 1 : génération et modélisation de l'environnement
- Injection 2 : simulation de phénomènes physiques
- Injection 3 : adaptation et interaction

4.2 . IA générative

Cette section présente le domaine de l'IA générative. Notre problème peut y être naïvement associé mais en réalité quasiment aucune des méthodes ne sera applicable. Les aspects présentés sont :

- Le concept d'IA générative. En notant que n'importe quelle fonction génère une sortie à partir d'une entrée et que la dérive de tout appeler IA générative est tentante.
- Les VAE et spécialement VAE conditionnels
- Les GAN et spécialement GAN conditionnels
- Les modèles de diffusion et spécialement ceux conditionnels
- Les modèles de langage et GPT

4.3 . Méthodes pour le traitement de séquence

Cette section présente les architectures connues pour leurs capacités à traiter des séquences, de leurs formes les plus simples aux formes les plus complexes. Par ordre d'apparition :

- Le concept de séquence : notion de proximité dans un ensemble. Série temporelle, image, texte.
- Réseau de convolution :
 - Histoire de son apparition : dans l'image
 - Comment la convolution interagit avec la séquence
 - La convolution dans l'image (vue comme une séquence)
 - La convolution dans le texte
 - La convolution ailleurs
- Réseau de neurones récurrents :
 - Histoire de son apparition
 - Comment un RNN interagit avec la séquence
 - Variante SSM

- RNN dans le texte
- RNN dans les systèmes temporels (chaîne de Markov)
- Transformer :
 - Histoire de son apparition : dans le langage
 - Comment le Transformer interagit avec la séquence
 - Transformer dans le texte (traduction, GPT, ...)
 - Transformer dans les systèmes temporels (chaîne de Markov)
 - Transformer dans l'image
 - Transformer ailleurs (généralisation)

4.4 . Les améliorations

Cette section met en avant les difficultés liées à l'apprentissage automatique, entre complexité calculatoire, mémorielle et instabilité en entraînement. À cette occasion, nous montrons les propositions existantes visant à résoudre ces problèmes. Par ordre d'apparition :

- Compréhension des architectures : Mechanistic Interpretability
- Présentation des soucis de performances
- Présentation des solutions aux soucis de performances
 - Positional Encoding
 - Certains mécanismes d'attention
 - Pre-Training
 - Embedding et tokenization
- Présentation des soucis de stabilité
- Présentation des solutions aux soucis de stabilité :
 - Layer-norm
 - Initialisation
 - Structure (hyper-paramètre de manière générale)
- Présentation des soucis d'efficacité et leurs solutions
 - Complexité mémoire et calcul : mécanisme d'attention
 - Vitesse d'entraînement : MAMBA?

5 - État de l'art

Ce chapitre présente les concepts et méthodes fondamentaux sur lesquels s'appuie cette thèse. Nous commencerons par définir la notion d'Environnement Numérique ainsi que les différents apports de l'IA à la simulation. Nous exposerons ensuite les approches d'IA générative, avant de nous concentrer sur les méthodes de traitement de séquence, en détaillant particulièrement l'architecture Transformer. Enfin, nous concluons par une analyse des défis inhérents à l'entraînement de ces modèles profonds, en passant en revue les solutions techniques existantes pour garantir leur stabilité, leur efficacité calculatoire et leur interprétabilité.

5.1 . Introduction : Environnements Numériques et typologie des apports de l'IA

L'émergence de l'industrie 4.0 et le développement rapide de l'intelligence artificielle ont propulsé l'utilisation de représentations virtuelles pour simuler, analyser et optimiser des systèmes physiques. Dans ce paysage technologique en pleine effervescence, les termes « jumeau numérique » et « environnement numérique » sont souvent employés de manière interchangeable, engendrant une confusion sémantique préjudiciable à la précision scientifique. Afin de poser les bases conceptuelles solides nécessaires à ce travail, cette section a pour objectif de démêler ces notions. Nous retracerons dans un premier temps l'origine et les définitions, tant idéales que pragmatiques, du jumeau numérique. Dans un second temps, nous présenterons une définition unificatrice et fonctionnelle de l'environnement numérique. Enfin, une synthèse comparative nous permettra d'établir une distinction claire basée sur les flux de données et le critère d'individualisation, et de justifier le positionnement terminologique adopté dans le cadre de cette étude.

5.1.1 . Cadre Conceptuel : Environnement Virtuel et Jumeau Numérique

Le concept de jumeau numérique, popularisé et formalisé dès le début des années 2000 par les travaux de Michael Grieves dans le domaine manufacturier [6], puis théorisé comme un pilier des systèmes cyber-physiques (CPS) par des auteurs comme Negri et al. [7], a connu une adoption rapide et variée à travers l'industrie.

Si le terme de "jumeau numérique" s'est imposé dans le paysage technologique, sa définition précise fait l'objet d'un débat animé entre une vision idéale et une approche pragmatique. D'un côté, les puristes, s'appuyant sur les travaux fondateurs de la NASA et de Grieves [6], défendent l'idée qu'un véritable jumeau numérique se caractérise par un couplage bidirectionnel et dynamique avec son homologue physique. Dans cette perspective exigeante, le jumeau n'est pas une simple représentation ; il est une représentation qui s'enrichit continuellement des données du physique et, en retour, pilote, optimise et prédit son comportement [7]. Cette boucle fermée est considérée comme la condition permettant de distinguer le jumeau numérique d'un simple modèle ou d'une simulation. De l'autre, une approche plus pragmatique, largement répandue dans l'industrie, adopte une définition évolutive et par niveaux de maturité. Dans cette

vision, une maquette 3D enrichie de données, parfois qualifiée de "digital shadow", peut déjà être labellisée "jumeau numérique". Cette flexibilité sémantique, bien que source de confusion, reflète la réalité des projets industriels où la complexité et le coût d'une intégration parfaite imposent une progression par étapes. Malgré tout, une ligne de démarcation essentielle fait consensus : l'existence d'un transfert de données automatique du système physique vers son représentant virtuel. Sans ce flux, la représentation demeure une simulation ou un modèle générique, que nous qualifierons ici d'« environnement numérique ». Par exemple les simulateurs de conduite autonome comme CARLA [8] sont des environnements numériques essentiels pour l'entraînement des algorithmes d'IA, mais ils simulent un monde routier générique non couplé à un véhicule physique unique, et ne sont en ce sens pas des jumeaux numériques. En revanche, certains simulateurs de moteur d'avion, comme ceux déployés par General Electric [9], qui est alimenté en temps réel par les données de vol de l'équipement spécifique, incarnent la définition minimale du jumeau numérique, souvent appelée « Digital Shadow ». Ils permettent un suivi individualisé de l'état de santé et de l'usure de chaque moteur de la flotte.

Pour désigner les représentations numériques qui ne sont pas couplées à une instance physique unique, nous recourons donc au terme plus large et unificateur d'Environnement Numérique (Virtual Environment - VE).

La notion d'Environnement Virtuel est interdisciplinaire, et sa définition varie selon que l'on se place dans la communauté de la Réalité Virtuelle, de l'Ingénierie Système ou de l'Intelligence Artificielle. La recherche en Réalité Virtuelle, historiquement focalisée sur l'immersion sensorielle et l'interaction humain-machine, définit souvent les VE comme des « mondes synthétiques générés par ordinateur dans lesquels l'utilisateur a un sentiment d'être présent et d'y interagir » [10]. Cette perspective met l'accent sur les aspects perceptuels et cognitifs. En revanche, dans les domaines de l'ingénierie et de l'IA, l'accent est davantage porté sur la fonction de simulation et de cadre d'expérimentation. Ici, un VE est vu comme un « modèle informatique exécutable d'un système » [11] ou un « cadre de simulation qui permet le test et la validation d'algorithmes dans des conditions contrôlées et reproductibles » [12]. Cette vision est moins concernée par l'immersion de l'utilisateur que par la fidélité de la modélisation des processus et des interactions.

Pour englober ces différentes finalités – de la formation immersive au banc d'essai algorithmique – nous proposons la définition unificatrice suivante : Un Environnement Virtuel (VE) désigne une simulation numérique interactive modélisant un ensemble d'entités et de phénomènes, dans le but d'observer, d'analyser ou d'expérimenter des comportements au sein d'un cadre contrôlé.

Cette définition permet de tracer une ligne de démarcation nette avec le concept voisin de Jumeau Numérique. La distinction fondamentale réside dans le principe d'individualisation par les données. Le jumeau numérique se définit intrinsèquement comme l'avatar d'une instance physique unique, tel un moteur spécifique, dont l'essence est indissociable d'un lien de données continu avec son homologue réel. En revanche, l'environnement numérique se conçoit comme une représentation générique d'une classe de systèmes, dont la valeur réside dans la modélisation fidèle de lois physiques au sein d'un cadre reproductible, indépendamment d'une instance matérielle particulière.

Ainsi clarifiée, la notion de VE couvre un spectre étendu, allant du monde immersif interactif au simulateur technique. Dans le contexte spécifique du développement algorithmique, qui est le nôtre, le VE devient un outil de prototypage et de validation indispensable : il permet de reproduire des situations expérimentales complexes, de générer des données synthétiques massives et de tester des modèles de manière intensive, sûre et économique, sans les contraintes logistiques des dispositifs physiques réels. Pour atteindre ce niveau de fidélité et d'efficacité, l'intégration de l'Intelligence Artificielle ne constitue pas une approche monolithique, mais se déploie selon trois axes d'intervention complémentaires que nous allons détailler : la création du contenu, l'accélération de la physique et l'adaptation décisionnelle.

5.1.2 . L'IA pour la constitution géométrique et visuelle de l'environnement

L'injection d'IA dans la conception des environnements numériques répond à deux impératifs distincts : la fidélité de la représentation du monde réel (modélisation) et la diversité des scénarios simulés (génération). Historiquement, ces tâches reposaient sur des processus manuels coûteux. L'apprentissage profond permet d'automatiser ces flux de travail à travers deux paradigmes complémentaires : la reconstruction neurale pour capturer le réel, et la synthèse générative pour créer des actifs inédits.

Modélisation : reconstruction neurale et représentations implicites

Le premier défi réside dans la numérisation automatisée d'environnements physiques existants. Les approches classiques de photogrammétrie, basées sur des maillages polygonaux explicites, montrent leurs limites en termes de gestion des reflets, de la translucidité et de la densité de stockage. Une alternative significative a émergé avec les représentations implicites, notamment les *Neural Radiance Fields* (NeRF) [13]. Cette méthode propose d'encoder la géométrie et l'apparence d'une scène non pas dans une structure de données géométrique, mais dans les poids d'un perceptron multicouche (MLP).

Bien que précis, le NeRF original présente des coûts d'entraînement et d'inférence élevés. L'introduction du *Hash Encoding* multi-résolution [14] a permis de réduire drastiquement les temps d'apprentissage. Plus récemment, une transition vers des représentations hybrides a été opérée avec le *3D Gaussian Splatting* [15]. En substituant le lancer de rayon volumétrique par une rasterisation de gaussiennes 3D anisotropes, cette méthode concilie la qualité visuelle des représentations neurales avec les performances temps réel (> 100 FPS) requises pour l'interactivité au sein du VE.

Génération : synthèse d'actifs par diffusion

Au-delà de la reproduction du réel, la simulation requiert la capacité de générer des environnements variés incluant des objets ou des conditions non observés. L'IA générative intervient ici pour la création d'actifs 3D, palliant la rareté des banques de modèles 3D par l'exploitation des vastes ensembles de données image-texte 2D.

L'état de l'art actuel s'appuie sur le transfert de connaissances depuis des modèles de diffusion 2D pré-entraînés vers la 3D. La méthode *DreamFusion* (Poole et al., 2022) [16] a formalisé ce principe via le *Score Distillation Sampling* (SDS). Cette technique utilise un modèle de diffusion 2D comme fonction de critique pour optimiser une représentation 3D (telle qu'un NeRF), de sorte que ses rendus 2D correspondent à une description textuelle donnée. Des itérations ultérieures, comme *ProlificDreamer* (Wang et al., 2023) [17], ont affiné ce processus via le *Variational Score Distillation* (VSD) pour améliorer la fidélité géométrique et la résolution des textures. Ces approches permettent d'envisager des pipelines de "texte-vers-environnement", où la description sémantique d'une scène suffit à instancier un cadre de simulation complet et physiquement cohérent.

5.1.3 . L'IA pour l'accélération et la modélisation des phénomènes physiques

L'objectif de cette seconde injection est de substituer ou d'accélérer les solveurs numériques traditionnels (Éléments Finis, Volumes Finis) dont la complexité calculatoire limite les applications temps réel. L'état de l'art s'articule autour de la manière dont la connaissance physique est intégrée dans le modèle d'apprentissage. Nous distinguons trois niveaux d'intégration, allant de l'apprentissage pur par les données à l'intégration structurelle des lois physiques.

Apprentissage par Observation (Data-Driven)

Le premier niveau considère le simulateur comme une "boîte noire" dont il faut approximer la fonction de transfert à partir d'observations. L'IA apprend ici les corrélations spatio-temporelles sans connaissance explicite des équations sous-jacentes. Les Graph Neural Networks (GNN) se sont imposés comme l'architecture de référence pour cette tâche, notamment pour les systèmes lagrangiens (particules). Les travaux sur les *Graph Network-based Simulators* (GNS) [18] démontrent une capacité remarquable à prédire la dynamique de fluides et de solides déformables en modélisant les interactions locales par passage de messages. Bien que très rapides à l'inférence, ces modèles souffrent d'un manque de garanties physiques : sans contrainte explicite, ils peuvent violer les lois de conservation (masse, énergie) et dériver sur de longues horizons temporels.

Apprentissage contraint par les Équations (Physics-Informed)

Pour pallier le manque de robustesse physique et la dépendance aux données, une seconde approche intègre les Équations aux Dérivées Partielles (EDP) directement dans l'optimisation. C'est le paradigme des Physics-Informed Neural Networks (PINNs) [19]. Ici, le réseau de neurones agit comme un approximateur universel de la solution, et sa fonction de coût inclut les résidus de l'équation physique (ex : Navier-Stokes). Cette méthode permet de s'affranchir partiellement ou totalement de données d'étiquetage (apprentissage non supervisé par la physique). Cependant, les PINNs font face à des défis d'optimisation majeurs lorsqu'ils sont confrontés à des dynamiques multi-échelles ou chaotiques.

Apprentissage structuré par la Physique (Inductive Bias)

Le troisième niveau d'intégration cherche à inscrire les lois physiques non plus dans la fonction de perte (contrainte douce), mais dans l'architecture même du réseau (contrainte dure ou biais inductif). D'une part, les Hamiltonian Neural Networks (HNN) [20] imposent une structure symplectique au réseau. Au lieu d'apprendre directement les accélérations, le réseau apprend l'Hamiltonien (l'énergie totale) du système, garantissant par construction la conservation de l'énergie et la réversibilité temporelle, ce qui est crucial pour la stabilité des simulations orbitales ou mécaniques sur le très long terme. D'autre part, les Fourier Neural Operators (FNO) [21] exploitent la structure spectrale des solutions d'EDP. En apprenant l'opérateur intégral dans l'espace de Fourier, ils acquièrent une propriété d'invariance à la discrétisation (zero-shot super-resolution), permettant de prédire la physique à des résolutions arbitraires, une propriété structurelle absente des CNN ou MLP classiques.

5.1.4 . L'IA au service de l'interactivité et de l'adaptation décisionnelle

Cette dernière dimension transforme l'environnement numérique d'un cadre passif en un écosystème réactif et adaptatif. L'objectif est d'enrichir la dynamique interactionnelle pour confronter le système sous test à des situations d'une complexité réaliste, impossible à coder manuellement via des scénarios déterministes.

L'environnement peuplé d'agents apprenants (IA comme Acteur)

La première contribution de l'IA est le remplacement des entités scriptées (PNJ, trafic, adversaires) par des agents autonomes pilotés par des politiques neuronales. Contrairement aux machines à états finis classiques, prévisibles et limitées, ces agents sont entraînés via l'Apprentissage par Renforcement Multi-Agents (MARL) ou des mécanismes de Self-Play [22]. Cela permet de peupler le VE d'adversaires ou de collaborateurs capables de stratégies émergentes et optimales. L'exemple du défi DARPA AlphaDogfight (2020), où des agents IA ont développé des manœuvres de combat aérien surclassant les experts humains, illustre comment l'injection d'agents apprenants permet de soumettre le système testé à des niveaux de difficulté et de réalisme inatteignables par des méthodes heuristiques [23]. Ici, l'environnement devient "intelligent" car ses composantes actives s'adaptent au comportement de l'utilisateur ou du système validé.

L'environnement comme générateur de curriculum (IA comme Superviseur)

La seconde contribution concerne le pilotage des paramètres de la simulation par des algorithmes d'optimisation ou évolutionnaires. Au-delà du simple Domain Randomization aléatoire [24], qui manque de direction, l'IA est utilisée pour structurer activement l'apprentissage : c'est l'Apprentissage de Curriculum Automatique (Automatic Curriculum Learning). Des méthodes comme POET (Paired Open-Ended Trailblazer) utilisent des algorithmes évolutionnaires pour co-générer l'environnement en même temps que l'agent [25]. L'algorithme cherche spécifiquement à générer les configurations topologiques ou physiques (terrains accidentés, conditions

météo limites) qui maximisent le progrès de l'agent, créant une "course à l'armement" entre la difficulté du monde et la compétence de l'agent. Dans ce cadre, les algorithmes évolutionnaires agissent comme une forme d'IA générative fonctionnelle, créant des scénarios pertinents et ciblés ("Edge cases") que le hasard seul ne produirait que rarement.

Ainsi, l'IA transforme le simulateur : d'un simple banc d'essai physique, il devient un partenaire d'entraînement actif, capable de générer des opposants redoutables et d'adapter sa propre complexité pour guider l'apprentissage.

5.1.5 . Ancrage dans la problématique

Au regard des distinctions établies précédemment, la classification du système étudié s'opère sans équivoque. Le simulateur de capteur de Mesures de Soutien Électronique que nous analysons a pour fonction de générer des données synthétiques à partir de scénarios tactiques génériques. Il ne modélise pas le comportement d'un équipement matériel spécifique connecté en temps réel, mais reproduit le fonctionnement théorique d'une classe de capteurs face à des environnements simulés. Par conséquent, l'absence de couplage à une instance physique unique et la nature générique des scénarios valident l'usage exclusif du terme d'Environnement Numérique (VE) pour désigner notre cadre d'étude.

L'analyse des trois axes d'intégration de l'IA révèle par ailleurs la singularité de notre approche au sein de ce VE. Bien que l'objectif fonctionnel rejoigne la "Seconde injection" visant l'accélération de la physique, les méthodes classiques sont inadaptées car elles traitent principalement des champs spatiaux continus. Or, notre goulot d'étranglement réside dans la transformation de séquences d'événements discrets (les PDW). Notre problématique se situe donc à l'intersection de la simulation physique et du traitement de l'information : il s'agit d'apprendre la fonction de transfert du capteur, un processus qui s'apparente conceptuellement à une tâche de "traduction" d'un état physique vers un état perçu. Ce constat motive l'adoption d'une approche fondée sur l'IA générative constructive et les architectures de traitement de séquence, dont nous explorerons les fondements théoriques dans les sections suivantes.

5.2 . IA générative

Dans le paysage contemporain de l'apprentissage profond, la définition de l'IA générative a évolué au-delà de la stricte opposition statistique entre modèles de densité et modèles discriminants. Là où un modèle classique condense l'information (classification, réduction de dimension), un modèle génératif apprend à construire des données de haute dimension, structurées spatialement ou temporellement, en capturant les dépendances complexes inhérentes au jeu d'entraînement. L'IA générative désigne aujourd'hui une classe d'architectures neuronales caractérisée par sa capacité de synthèse.

Un modèle est qualifié de génératif dès lors qu'il construit une donnée structurée en capturant la distribution de probabilité sous-jacente. L'objectif n'est pas simplement d'estimer une

valeur locale, mais de bâtir une cohérence globale, respectant les corrélations intrinsèques du domaine d'apprentissage. Cette définition par la capacité constructive est particulièrement pertinente pour la modélisation de systèmes physiques, où la distinction entre discret et continu s'estompe

Dans le contexte spécifique de l'accélération de simulation, nous nous intéressons particulièrement aux modèles génératifs conditionnels, capables de produire une sortie structurée complexe, tel un champ physique ou un état futur, correspondant à une condition initiale. Cette section explore l'évolution chronologique de ces architectures, depuis les approches opérant dans des espaces continus jusqu'aux paradigmes séquentiels discrets.

5.2.1 . L'approche probabiliste explicite : Les VAE (2013)

La première avancée significative dans l'apprentissage profond de distributions complexes fut l'introduction des Auto-encodeurs Variationnels (VAE). Contrairement aux auto-encodeurs classiques qui compresse l'information en un point déterministe de l'espace latent, les VAE imposent une structure probabiliste à cet espace, généralement sous la forme d'une distribution gaussienne multivariée. L'innovation majeure réside dans l'introduction de l'astuce de reparamétrisation (reparameterization trick), qui rend le processus d'échantillonnage différentiable et permet l'optimisation du modèle par descente de gradient en maximisant la borne inférieure de la vraisemblance (ELBO) [26]. Cette capacité à structurer l'espace latent est particulièrement pertinente pour les problèmes de simulation où une même condition initiale peut mener à plusieurs résultats possibles. Dans leur article sur les VAE Conditionnels (C-VAE) [27], il est prouvé qu'il est possible de modéliser des sorties structurées multimodales en conditionnant la génération à la fois par une variable latente aléatoire et par une observation d'entrée. Bien que théoriquement élégants, les VAE ont souffert historiquement d'une limitation qualitative, leur fonction de perte tendant à produire des résultats lissés. Cependant, des développements récents ont redonné une pertinence majeure à cette famille, notamment via la quantification vectorielle de l'espace latent (VQ-VAE). Ces modèles discrets sont désormais au cœur d'architectures de pointe comme les World Models, où un agent apprend à "rêver" des futurs possibles dans un espace latent compact pour accélérer l'apprentissage par renforcement en robotique [28], [29].

5.2.2 . La révolution antagoniste : Les GAN (2014)

Pour répondre au manque de piqué et de réalisme des méthodes variationnelles, une rupture paradigmatique a été introduite avec les Réseaux Antagonistes Génératifs (GAN). Cette approche délaisse l'estimation explicite de la densité de probabilité au profit d'une méthode implicite fondée sur la théorie des jeux. Le processus d'apprentissage est modélisé comme un jeu minimax à somme nulle entre un générateur qui tente de créer des données indiscernables du réel, et un discriminateur qui tente de distinguer les échantillons générés des données d'entraînement [30]. Comme le soulignent les travaux théoriques sur les modèles implicites, cette formulation permet au générateur d'apprendre des statistiques d'ordre supérieur souvent ignorées par les méthodes classiques, s'affranchissant des contraintes de vraisemblance [31]. Dans le cadre de la "traduction" d'environnement, les variantes conditionnelles telles que l'architec-

ture Pix2Pix se sont imposées pour transformer une représentation sémantique en une image photoréaliste, produisant des structures fines et des textures détaillées [32]. Si les GAN restent difficiles à stabiliser durant l'entraînement, ils ont démontré des capacités de généralisation spectaculaires au-delà de l'image statique. Des travaux comme tempoGAN ont par exemple appliqué ce principe à la mécanique des fluides, parvenant à super-résoudre des simulations volumétriques de fumée ou de liquide tout en garantissant une cohérence temporelle que les méthodes purement statistiques peinent à maintenir [33].

5.2.3 . La génération par raffinement : Les Modèles de Diffusion (2020)

La dernière vague d'innovation, qui définit une grande partie de l'état de l'art actuel, puise son inspiration dans la physique statistique hors équilibre. Les modèles de diffusion probabilistes proposent de construire la génération comme l'inversion d'un processus de destruction d'information. L'idée consiste à détruire progressivement la structure des données par l'ajout successif de bruit gaussien, puis d'entraîner un réseau de neurones à inverser ce processus temporel pour reconstruire la donnée originale étape par étape [34]. Ce concept a été porté à maturité avec les Denoising Diffusion Probabilistic Models (DDPM), qui offrent un compromis inédit : ils atteignent une qualité d'échantillonnage supérieure aux GAN tout en couvrant mieux la diversité de la distribution des données, évitant le problème d'effondrement de mode [35]. Bien que le processus itératif soit intrinsèquement lent, des méthodes d'échantillonnage accélérées (DDIM) ont rendu ces modèles exploitables en production [36]. Au-delà de la génération d'images 2D, ce paradigme est aujourd'hui le moteur de la génération d'actifs pour les environnements virtuels. Des approches comme DreamFusion utilisent un modèle de diffusion 2D pré-entraîné pour optimiser une représentation volumétrique (NeRF), permettant de générer des objets 3D complets et cohérents à partir d'une simple description textuelle, ouvrant la voie à la création procédurale d'environnements physiques complexes [16].

5.2.4 . Le paradigme séquentiel et l'Autorégression

Enfin, une approche radicalement différente considère la génération comme une prédiction séquentielle discrète. Ce paradigme trouve ses racines dans les Réseaux de Neurones Récurrents (RNN), historiquement utilisés pour générer du texte ou des séries temporelles, mais limités par leur mémoire à court terme et leur séquentialité stricte [37]. La rupture fondamentale survient avec l'introduction de l'architecture Transformer et du mécanisme d'attention, qui permet de modéliser des dépendances à très long terme et de paralléliser le calcul [38]. L'évolution majeure de ce paradigme réside dans le concept de pré-entraînement génératif (GPT). Il a été démontré qu'un modèle entraîné massivement sur l'objectif simple de prédire le prochain élément d'une séquence acquiert une capacité de généralisation et de compréhension structurelle émergente [39]. Aujourd'hui, cette approche déborde largement du cadre du texte. Des architectures multimodales comme Gato [40] ou les Vision Transformers (ViT) [41] traitent les images ou les actions de contrôle robotique comme des séquences de tokens, unifiant ainsi la génération de contenu visuel et la prise de décision séquentielle au sein d'un même formalisme autorégressif. Cela positionne le traitement de séquence comme une méthode universelle pour

la simulation, justifiant l'analyse détaillée des architectures séquentielles qui suivra.

5.2.5 . Ancrage dans la problématique

L'analyse du paysage de l'IA générative nous permet d'identifier les architectures les plus adaptées à la modélisation de notre simulateur de capteur. Si les modèles probabilistes explicites comme les VAE offrent une gestion intéressante de l'incertitude structurelle, leur tendance historique à produire des sorties lissées peut poser question quant à la fidélité des signaux radar, où la précision des paramètres fins tels que les fréquences est critique. Concernant les approches antagonistes (GAN), bien qu'elles soient performantes pour la synthèse d'images, leur adaptation directe à des séquences d'événements discrets paramétriques (les PDW) s'avère complexe, notamment pour gérer la causalité temporelle et la nature irrégulière du flux d'impulsions, sans compter leur instabilité d'entraînement connue. De même, si les modèles de diffusion définissent l'état de l'art actuel, leur processus de débruitage itératif est intrinsèquement coûteux en temps de calcul. Cette caractéristique entre potentiellement en conflit avec notre objectif premier d'accélération de la simulation, en plus de nécessiter une adaptation lourde pour traiter des vecteurs de paramètres physiques plutôt que des pixels.

Cette analyse invite à considérer le paradigme séquentiel auto-régressif. Contrairement au Traitement du Langage Naturel qui opère sur des vocabulaires finis, notre simulation numérique évolue dans un espace métrique continu : chaque PDW est défini par des coordonnées réelles (temps d'arrivée, fréquence, largeur). Dans ce contexte, l'acte génératif ne consiste pas à sélectionner un symbole parmi un dictionnaire, mais à prédire directement les valeurs d'un état dans un espace vectoriel continu \mathbb{R}^n . Bien que ce processus s'apparente mathématiquement à une régression multivariée, il conserve la nature intrinsèque de la génération : le modèle doit bâtir, étape par étape, une cohérence globale du signal temporel. Ainsi, la reconstruction de la séquence de PDW perçue à partir de la séquence émise se formule comme une tâche de traduction de signal continu. Cela motive l'orientation de notre étude vers les architectures spécialisées dans le traitement de séquence, capables de capturer les dépendances à long terme comme le pistage temporel, justifiant l'analyse approfondie des RNN et des Transformers qui suivra.

5.3 . Méthodes de traitement : Séquences et structures spatiales

5.3.1 . Typologie des données : De la causalité temporelle à la topologie spatiale

Avant d'aborder les architectures neuronales spécifiques, il est essentiel de définir formellement l'objet mathématique qu'elles manipulent. Dans l'apprentissage statistique classique, les données sont souvent supposées être indépendantes et identiquement distribuées (hypothèse i.i.d.). Le traitement de séquences et de structures spatiales rompt fondamentalement avec cette hypothèse en introduisant une structure de dépendance intrinsèque. Une donnée n'est pas un simple ensemble non ordonné, mais une collection indexée $X = \{x_1, x_2, \dots, x_N\}$ où l'indice porte une information topologique ou causale déterminante. La valeur d'un élément x_i n'a de sens que relativement à son contexte, c'est-à-dire son voisinage ou son historique.

Cette propriété de dépendance locale, formalisée par les processus de Markov pour le temps ou les Champs de Markov pour l'espace, caractérise la nature de l'ensemble et impose des contraintes spécifiques de modélisation que les architectures neuronales doivent satisfaire.

La séquence et la causalité

Dans le cadre des séquences, la notion de proximité est dictée par la causalité : l'état présent est une fonction de l'histoire passée. C'est le fondement de la théorie de l'information de Shannon [42], où le langage est modélisé comme un processus stochastique discret. Dans cette vision, la probabilité d'apparition d'un symbole (lettre ou mot) dépend conditionnellement de la séquence des symboles précédents, définissant la notion d'entropie d'une source d'information. Cette logique s'applique identiquement aux séries temporelles continues. Des travaux sur les modèles ARIMA [43] ont montrés qu'une observation à l'instant t est mathématiquement corrélée à ses prédécesseurs immédiats et aux termes d'erreur passés. Dans ce formalisme statistique, la séquence est définie par une dépendance directionnelle irréversible vers le futur.

La donnée spatiale et la contiguïté

Le champ physique ou l'image, de leur côté, se définissent comme des grilles spatiales (multidimensionnelle ou simplement bidimensionnelle) où la notion d'ordre séquentiel disparaît au profit de la contiguïté topologique. Ici, un pixel ou une cellule n'a pas de "passé" ou de "futur", mais un voisinage omnidirectionnel. Cette structure, régie par la contiguïté spatiale locale, s'affranchit de la causalité temporelle au profit d'un voisinage omnidirectionnel propre aux Champs de Markov. Elle permet de fonder théoriquement l'usage des opérations de convolution, où la notion d'ordre chronologique est remplacée par celle de proximité euclidienne. Ce concept de donnée spatiale reste néanmoins proche de celui de séquence : les travaux sur PixelRNN [44] montre qu'en traitant l'image comme une séquence autorégressive, où chaque pixel dépend de ceux situés "avant" lui (en haut et à gauche), on peut modéliser la distribution conjointe des pixels et générer des structures visuelles cohérentes.

Universalité de la modélisation séquentielle

Finalement, le défi central des architectures de traitement que nous allons présenter (CNN, RNN, Transformer) est de modéliser cette fonction de dépendance conditionnelle $P(x_t | \text{Contexte})$. La nature de ce contexte varie selon le domaine : il sera un historique causal pour les séries temporelles et la génération de texte, alors qu'il est bidirectionnel et topologique pour l'image ou la compréhension sémantique globale. Cependant, l'objectif mathématique reste identique : capturer les corrélations à courte et longue portée qui structurent la donnée, transformant une collection de valeurs isolées en une entité cohérente.

5.3.2 . Réseaux de convolution

Bien que les données séquentielles soient intuitivement associées à une dimension temporelle linéaire, le traitement de l'information repose fondamentalement sur l'extraction de

motifs locaux et de relations de voisinage. C'est dans cette optique que les Réseaux de Neurones Convolutionnels (CNN) se positionnent comme une méthode incontournable. Initialement conçus pour la grille spatiale de l'image, ils formalisent une approche du traitement de séquence fondée sur la localité, l'invariance par translation et la hiérarchie des caractéristiques.

Genèse et prédominance dans l'imagerie

L'histoire des réseaux de convolution est indissociable de la vision par ordinateur et de la volonté de s'affranchir des descripteurs manuels (SIFT [45], SURF [46] et HOG [47]). Inspirée par les travaux biologiques sur le cortex visuel, le premier modèle [48] a introduit les concepts fondateurs de champ récepteur local et de partage des poids pour la reconnaissance de caractères manuscrits. Cependant, c'est l'avènement d'AlexNet [49] qui a marqué le véritable point d'inflexion en démontrant la supériorité de l'apprentissage profond sur GPU pour l'extraction de caractéristiques. Cette percée a ouvert la voie à des architectures plus profondes et plus efficaces. Par exemple, l'architecture GoogLeNet [50] factorise les convolutions pour réduire le coût de calcul tout en augmentant la largeur du réseau, permettant de traiter des motifs à différentes échelles simultanément.

Mécanisme d'interaction : Filtrage local et expansion hiérarchique

L'interaction fondamentale d'un réseau de convolution avec une séquence repose sur l'application répétée d'un opérateur de filtrage caractérisé par un noyau w de support fini. Contrairement à une couche dense qui apprendrait un poids spécifique pour chaque élément de la séquence globale, la convolution impose une contrainte de partage des poids qui nécessite que les données soient structurées dans un espace métrique régulier. En effet, l'opération pré-suppose l'existence d'une fonction $p(\cdot)$ permettant d'associer à chaque élément x sa position sur une grille sous-jacente, qu'elle soit unidimensionnelle pour des données temporelles ou multidimensionnelle pour des données spatiales.

Mathématiquement, le noyau w est défini sur un support \mathcal{V} centré à l'origine. Ainsi, le noyau définit pour tout élément cible x_t un voisinage d'interaction \mathcal{V}_t , correspondant à la translation du support \mathcal{V} en la position $p(x_t)$. L'opération de convolution consiste alors à calculer une somme pondérée des éléments appartenant à ce voisinage, où les poids sont déterminés exclusivement par la position relative entre les éléments et le centre. La sortie h_t (avant activation) s'exprime par l'équation :

$$h_t = \sum_{x_j \in \mathcal{V}_t} w_{\Delta(x_j, x_t)} \cdot x_j + b$$

Dans cette expression, $\Delta(x_j, x_t) = p(x_j) - p(x_t)$ représente le vecteur de position relative du voisin x_j par rapport au centre x_t , b est un biais. Une conséquence directe de la structure en grille des données est que ce vecteur de différence correspond systématiquement à un n-uplet d'entiers (n étant la dimension de la grille). Cette propriété discrète est fondamentale pour l'implémentation neuronale : elle implique que la fonction w n'a pas besoin d'être modélisée comme une fonction continue, mais se réduit à un ensemble fini de paramètres scalaires (les poids du

filtre) qu'il suffit d'apprendre pour chaque décalage entier possible dans le support. Cette formulation garantit l'invariance par translation, assurant que le même motif de poids est appliqué uniformément sur toute la structure. Par ailleurs, l'application de ce voisinage aux bornes d'une grille finie nécessite une gestion des effets de bord, typiquement résolue par l'ajout de valeurs nulles (zero-padding) en périphérie afin de conserver la dimension de la séquence traitée.

Ce mécanisme permet l'extraction robuste de motifs locaux, mais la compréhension de la structure globale de la séquence émerge de la composition hiérarchique de ces opérations. L'illustration 5.1 permet de visualiser comment l'empilement de couches induit une expansion mathématique du champ récepteur. Considérons une première couche définie par un filtre w de taille 3. Pour un instant t , ce filtre induit un voisinage immédiat. L'équation locale est, en notant ϕ la fonction d'activation :

$$\begin{aligned} h_t^{(1)} &= w_1 x_{t-1} + w_2 x_t + w_3 x_{t+1} \\ x_t^{(1)} &= \phi(h_t^{(1)}) \end{aligned}$$

La sortie $x_t^{(1)}$ est une fonction des entrées x_{t-1} à x_{t+1} . Lorsqu'une seconde couche définie par un filtre v de même support est appliquée sur cette représentation intermédiaire, elle opère selon le même principe d'invariance en translation :

$$\begin{aligned} h_t^{(2)} &= v_1 x_{t-1}^{(1)} + v_2 x_t^{(1)} + v_3 x_{t+1}^{(1)} \\ x_t^{(2)} &= \phi(h_t^{(2)}) \end{aligned}$$

La sortie $x_t^{(2)}$ est une fonction des entrées x_{t-2} à x_{t+2} . Ainsi, par simple composition algébrique, l'horizon d'interaction s'est étendu de 3 éléments (couche 1) à 5 éléments (couche 2). La profondeur du réseau agit donc comme un multiplicateur mécanique de l'horizon d'interaction, permettant de reconstruire des dépendances causales à longue portée à partir de règles de construction strictement locales et invariantes.

Au-delà de l'expansion de l'horizon d'interaction, la géométrie du support de convolution détermine la nature causale ou non du traitement, une caractéristique nécessaire pour la modélisation de systèmes dynamiques. La partie gauche de la figure 5.2 la configuration standard, dite convolution centrée. Pour calculer un élément de sortie y_4 à l'instant $t = 4$, le champ récepteur effectif (cône violet) agrège les informations d'un voisinage symétrique de l'entrée x , de x_2 à x_6 . Cette approche est naturelle pour l'analyse de données statiques (comme une image) ou le traitement de séquences complètes a posteriori, mais n'est pas adapté à la modélisation d'un système dynamique. Par exemple dans le cas de filtrage en ligne, le système ne peut physiquement pas accéder aux mesures futures pour débruiter les données du présent. Pour adapter l'architecture à ces contraintes temporelles strictes, on recourt à la convolution causale. Cette variante consiste à décaler le support du filtre de manière à ce que le voisinage d'interaction au temps t ne contienne aucun indice supérieur à t . L'exemple illustré sur la partie droite de la figure 5.2 assure que le cône d'influence de la sortie y_5 est alors strictement orienté vers le passé (x_1 à x_5). Dans cette configuration, le réseau conserve sa capacité d'extraction de motifs et de

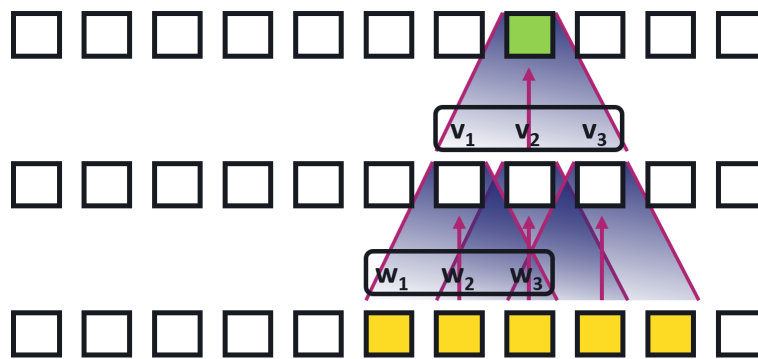


Figure 5.1 – Illustration d’une convolution 1D standard et de l’expansion hiérarchique du champ récepteur

parallélisation, mais adopte une topologie d’interaction compatible avec la physique, simulant le comportement d’un système causal sans recourir à la mémoire récurrente.

La convolution dans l’image : Une séquence spatiale 2D

Dans le contexte de l’image, la séquence est bidimensionnelle et le CNN y opère une extraction hiérarchique. Les premières couches détectent des primitives simples comme des bords ou des textures, qui sont ensuite combinées pour former des motifs sémantiques complexes. Cette capacité d’abstraction a été poussée à son paroxysme par l’architecture VGG [51], qui a standardisé l’usage de filtres de très petite taille (3×3) empilés en grande profondeur. Les auteurs ont démontré qu’une séquence de petites convolutions est plus efficace pour capturer des non-linéarités complexes qu’une seule grande convolution. Cependant, l’augmentation de la profondeur a engendré des problèmes de disparition du gradient, résolus avec ResNet [52]. L’introduction de connexions résiduelles a permis d’entraîner des réseaux dépassant la centaine de couches, essentiels pour capturer les dépendances à très longue portée dans des images haute résolution. Pour les tâches de "traduction" d’image vers image, cruciales en simulation (par exemple, passer d’une carte de densité à un champ de pression), il est impératif de ne pas perdre l’information spatiale lors de la compression. L’architecture U-Net [53], initialement pour la segmentation biomédicale combine un chemin de contraction et un chemin d’expansion reliés par des connexions latérales (skip connections). Cette structure permet de générer une sortie de même résolution que l’entrée en fusionnant le contexte sémantique global et les détails locaux. Cette architecture est aujourd’hui une référence pour les modèles de substitution en physique. D’autres variantes comme DenseNet [54] ont poussé cette logique plus loin en connectant chaque couche à toutes les suivantes pour maximiser le flux d’information, bien que cela se fasse au prix d’une consommation mémoire accrue.

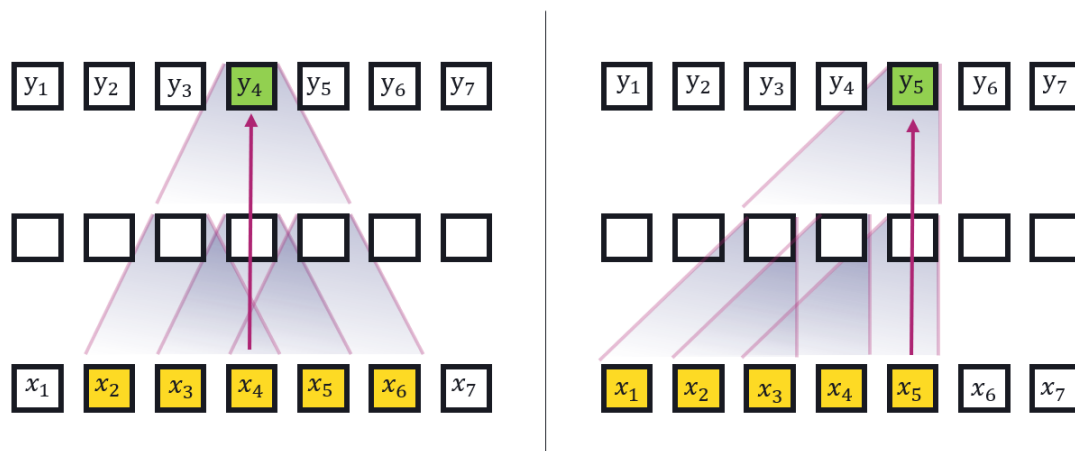


Figure 5.2 – Impact de la topologie du support de convolution sur la causalité temporelle : approche centrée (gauche) et approche causale (droite)

La convolution dans les séquences 1D (Texte, Audio, Séries Temporelles)

Bien que souvent associés à l'image, les CNN se sont révélés extrêmement performants pour traiter des séquences unidimensionnelles, surpassant parfois les réseaux récurrents grâce à leur capacité de parallélisation. Dans le traitement du signal audio, l'architecture WaveNet [55] a marqué une rupture en utilisant des convolutions causales dilatées. Ce mécanisme permet au champ récepteur du réseau de croître exponentiellement avec la profondeur sans augmenter le nombre de paramètres, capturant ainsi des dépendances temporelles sur des milliers de pas de temps, ce qui est impossible pour un RNN standard REF. Dans le domaine du traitement du langage naturel (NLP), cette logique a été appliquée avec succès à la traduction automatique. L'architecture ConvS2S [56] entièrement convolutionnelle pour la séquence à séquence, utilise des mécanismes d'attention multi-pas pour pondérer l'importance des mots sources. De même, ByteNet [57], réalise la traduction en temps linéaire en empilant des convolutions dilatées. Ces travaux ont démontré que l'induction de biais locaux propres aux convolutions est pertinente pour la syntaxe et la sémantique locale. Cette approche a été généralisée aux séries temporelles génériques sous le nom de Temporal Convolutional Networks (TCN) [58]. L'étude comparative démontre que sur une vaste gamme de tâches séquentielles, comme la prédiction de charge énergétique ou la modélisation de séquences symboliques, les TCN surpassent souvent les réseaux récurrents (LSTM/GRU) REF tout en offrant une stabilité d'entraînement supérieure. Une étude récente [59] prolonge ce constat en suggérant que des architectures convolutionnelles modernes pré-entraînées peuvent rivaliser avec les Transformers sur certaines tâches textuelles, soulignant la pertinence continue de ce paradigme.

Généralisation : De la grille volumétrique aux topologies irrégulières

Le principe de convolution, initialement restreint aux images planes, a fait l'objet d'extensions successives pour traiter des structures de données de plus en plus complexes. Une première étape de généralisation concerne les données volumétriques (3D) et spatio-temporelles, qui conservent toutefois une structure de grille régulière (euclidienne). Pour l'analyse de vidéos, C3D [60] déploie des filtres tridimensionnels (x, y, t) capturant conjointement l'espace et le mouvement. De même, dans le domaine médical, l'architecture V-Net [61] étend le principe du U-Net à la 3D en opérant sur des voxels. Bien que performantes, ces méthodes restent contraintes par la régularité de la grille : elles imposent une voxelisation des données qui induit une complexité cubique et une perte de résolution, les rendant inadaptées aux géométries éparses.

La généralisation la plus significative pour la simulation scientifique concerne donc les données non-euclidiennes, intrinsèquement irrégulières. Dans une simulation lagrangienne ou un système moléculaire, le "voisinage" n'est plus défini par une position dans une matrice, mais par la topologie d'un graphe. Les Graph Convolutional Networks (GCN) [62] redéfinissent alors la convolution comme une agrégation spectrale ou spatiale des caractéristiques des nœuds connectés, indépendamment de leur disposition géométrique absolue. Cette approche a été enrichie par GraphSAGE [63], proposant une convolution inductive capable de généraliser à des nœuds invisibles durant l'entraînement. Enfin, pour traiter des nuages de points 3D bruts sans passer par la case voxelisation, des architectures comme PointNet++ [64] appliquent des opérations hiérarchiques directement sur des ensembles continus, comblant le fossé entre convolution discrète et géométrie continue.

5.3.3 . Réseaux de neurones récurrents et Espaces d'Etats (RNN et SSM)

Si les réseaux de convolution abordent la séquence par une fenêtre glissante locale, une autre famille d'architectures adopte une approche intrinsèquement temporelle : la modélisation récurrente. Qu'il s'agisse des Réseaux de Neurones Récurrents (RNN) historiques ou des récents Modèles d'Espaces d'États (SSM), le principe fondateur reste la persistance de l'information. Le modèle maintient un état caché interne h_t qui agit comme une mémoire compressée de tout l'historique passé, mise à jour à chaque nouvelle observation. Cette formulation est particulièrement naturelle pour la simulation physique, car elle mime la dynamique des systèmes causaux où l'état futur dépend de l'état présent et des forces appliquées.

Genèse et mécanismes d'interaction : De la boucle simple aux portes logiques

L'histoire de cette approche débute avec les RNN classiques [65] qui introduisent une boucle de rétroaction permettant au réseau de maintenir une trace du contexte temporel. Cependant, bien que ces réseaux parviennent à générer des séquences continues complexes comme de l'écriture manuscrite, ils souffrent d'une instabilité critique lors de l'entraînement : le problème de la disparition ou de l'explosion du gradient [37]. Sur de longues séquences, le signal d'erreur se dilue, empêchant l'apprentissage des causes lointaines d'un événement. Pour y remédier, le LSTM (Long Short-Term Memory) [66] propose des "cellules" mémoires protégées par des

portes logiques, et peut choisir de retenir ou d'effacer une information sur des milliers de pas de temps. Cette capacité a été affinée par l'introduction du GRU [67], [68], une variante plus économe.

Mécanisme d'interaction : Récurrence et Mémoire d'État

L'interaction fondamentale des architectures récurrentes (RNN) et des modèles d'espaces d'états (SSM) avec la séquence repose sur un principe de persistance de l'information, radicalement différent de la localité spatiale des convolutions. Au lieu d'agréger un voisinage statique, ces modèles introduisent une variable latente dynamique, l'état caché h_t , qui agit comme une mémoire compressée de l'historique causal. Mathématiquement, la modélisation d'une telle trajectoire dynamique se formalise comme un problème de valeur initiale. Elle nécessite impérativement de définir une condition d'ancrage h_0 avant de décrire la loi d'évolution f . Pour une séquence d'entrée x , le système est ainsi entièrement décrit par le couple d'équations suivant :

$$\begin{cases} h_0 = \lambda \\ h_t = f(h_{t-1}, x_t; \theta) \quad \forall t \geq 1 \end{cases}$$

La première équation initialise l'état de la mémoire avant toute observation. Si la convention standard pose un vecteur nul ($\lambda = \vec{0}$), supposant une mémoire vierge, l'apprentissage machine laisse la possibilité au réseau d'optimiser l'encodage de cette mémoire vierge sous la forme d'un paramètre appris λ . La seconde équation décrit la mise à jour récursive où f est une fonction de transition paramétrée par θ . Cette formulation implique que h_t ne dépend pas seulement de l'entrée locale x_t , mais indirectement de toute la trajectoire passée $\{x_0, \dots, x_t\}$ accumulée dans h_{t-1} .

En pratique, cette dynamique de mise à jour est régie par un ensemble de paramètres aprenables qui sont partagés sur des parties de la séquence, garantissant l'invariance du traitement par tâche conceptuelle (genre encodage d'une phrase puis autre matrice pour décodage et traduction). Historiquement, les premières architectures (genre RNN avant LSTM et GRU etc) encodaient simplement ces paramètres sous la forme d'une matrice W_{ih} (Input-to-Hidden) qui projette l'entrée courante dans l'espace latent, d'une matrice W_{hh} (Hidden-to-Hidden) qui contrôle l'évolution de la mémoire interne, et d'un vecteur de biais b . En notant ϕ la fonction d'activation non-linéaire, l'équation de propagation de récurrence s'écrit pour notre exemple :

$$h_{t+1} = \phi(W_{ih}x_t + W_{hh}h_t + b)$$

L'illustration 5.3 permet de visualiser la propagation de la dépendance temporelle à travers une architecture récurrente élémentaire, souvent qualifiée de RNN d'Elman [65]. Dans ce modèle simplifié (ici représenté sans biais pour la clarté), les informations sont transportées d'état caché en état caché à travers la matrice de poids W_1 et d'entrée à l'état caché à travers W_2 . On illustre en bleu le cône de perception de l'état caché h_3 (représenté par le carré vert) intégrant par récurrence les informations des observations passées x_1 , x_2 et x_3 , créant un lien causal

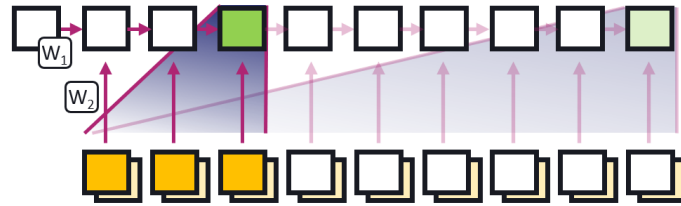


Figure 5.3 – Mécanisme fondamental de la récurrence et propagation de l'état mémoire

ininterrompu.

$$\begin{aligned}
 h_3 &= \phi(W_2x_3 + W_1h_2) \\
 &= \phi(W_2x_3 + W_1\phi(W_2x_2 + W_1h_1)) \\
 &= \phi(W_2x_2 + W_1\phi(W_2x_2 + W_1\phi(W_2x_1 + W_1\lambda)))
 \end{aligned}$$

Cette formulation met en évidence la différence fondamentale avec la convolution : alors que le champ récepteur d'un CNN s'élargit progressivement par empilement de couches, ici le cône d'influence (en bleu) s'étend horizontalement vers le passé jusqu'à la première entrée, capturant théoriquement la totalité de l'historique causal disponible.

En fonction de la fonction accomplie par l'architecture récurrente, les états cachés sont exploités différemment. Le mode "Flux à Flux" (ou Many-to-Many synchronisé), illustré à gauche dans la figure 5.4, aligne la production de la sortie sur la réception de l'entrée. À chaque pas de temps t , l'état caché h_t est utilisé immédiatement pour prédire une sortie y_t . Cette configuration, où la causalité est stricte et le délai minimal, est caractéristique des systèmes de filtrage en ligne ou de contrôle, où la réaction doit être instantanée. Le mode "Séquence vers Séquence" (Seq2Seq), à droite dans la figure 5.4, nécessite d'opérer en deux phases distinctes. La phase d'encodage représenté en violet, ingurgite toutes les informations et les condense dans l'état caché h_5 . Les états cachés h_1 à h_5 ne sont utilisés que pour encoder l'information du passé, et ne sont exploités que pour transmettre cette information à l'avenir. La phase de décodage récupère ce contexte passé pour générer la séquence de sortie. Les états cachés de h_6 à h_8 sont à la fois utilisés pour transmettre et renseigner l'information de mémoire mais aussi pour prédire les nouveaux éléments, qui sont réinjectés en entrée (boucle autorégressive). Ce mécanisme permet de transformer une séquence en une autre de longueur différente et de modéliser des dépendances non-monotones, mais impose que toute l'information pertinente soit compressée dans un goulot d'étranglement. C'est cette distinction topologique, plus que la nature des données, qui différencie fondamentalement l'usage des RNN pour le suivi de signal (mode flux) de leur usage pour la traduction (mode Seq2Seq).

Renouveau architectural : Les Modèles d'Espaces d'Etats (SSM)

Malgré leur robustesse, les LSTM conservent une limitation structurelle majeure : leur traitement séquentiel interdit la parallélisation sur GPU. C'est pour lever ce verrou qu'une nou-

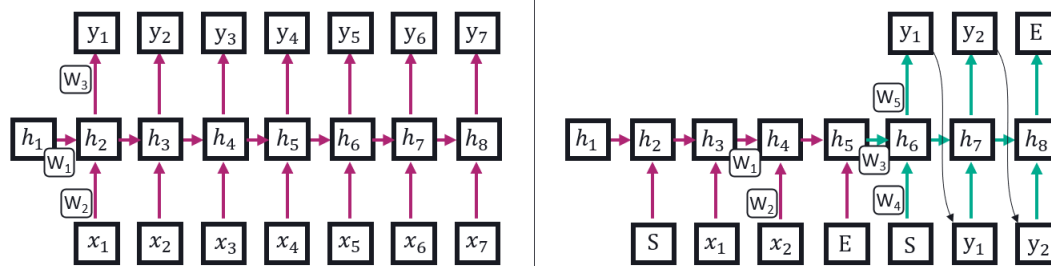


Figure 5.4 – Topologies d'application des architectures récurrentes : traitement de flux (gauche) et traduction globale (droite)

velle classe de modèles a émergé : les State Space Models (SSM). Ces modèles puisent leur origine théorique dans le papier HiPPO [69], qui formalise mathématiquement comment compresser optimalement une histoire continue dans un vecteur de taille fixe via des projections polynomiales orthogonales. Cette base a permis de développer S4 (Structured State Space sequence model) [70], capable de modéliser des dépendances sur plus de 10 000 pas de temps en résolvant une équation différentielle continue discrétisée. Cependant, les premiers SSM souffraient d'une rigidité dynamique, peinant à sélectionner l'information pertinente en fonction du contexte ("Content-based selection"). Cette limite a été adressée par l'architecture Mamba [71]. En rendant les matrices d'état dépendantes de l'entrée, Mamba atteint des performances comparables aux premiers Transformers REF tout en conservant une complexité linéaire. Toutefois, ces modèles restent délicats à stabiliser sur des dynamiques hautement chaotiques où la discrétisation numérique peut introduire des dérives.

Application au texte : L'ère du Sequence-to-Sequence et de la Traduction

Dans le traitement du langage, l'approche récurrente a connu son apogée avec le paradigme Seq2Seq [72]. En utilisant deux LSTM (un encodeur et un décodeur), cette architecture a permis de traiter des séquences de longueurs variables. Cette avancée a transformé l'industrie de la traduction avec le déploiement du Google's Neural Machine Translation System (GNMT) [73] en 2016, réduisant les erreurs de traduction de près de 60 % par rapport aux systèmes statistiques. Au-delà de la traduction, ce paradigme a permis des avancées dans la modélisation prédictive de parcours complexes, comme l'illustre par le modèle Doctor AI [74]. Ce modèle utilise des RNN pour prédire les futurs diagnostics médicaux et la durée avant la prochaine visite à partir de l'historique clinique des patients, démontrant la capacité des RNN à capturer des dynamiques temporelles irrégulières et multivariées dans des données réelles bruitées.

Application aux systèmes temporels, physiques et créatifs

Au-delà du texte, les RNN se sont imposés comme l'outil naturel pour la modélisation de systèmes dynamiques continus, un domaine crucial pour la simulation. Une étude sur la prévision de systèmes chaotiques [75] a mis en avant que les LSTM pouvaient apprendre la dynamique de l'attracteur de Lorenz ou de l'équation de Kuramoto-Sivashinsky mieux que les modèles physiques simplifiés, en capturant les propriétés non-linéaires de l'évolution temporelle à court terme. Cette capacité à modéliser le chaos déterministe fait des RNN des candidats sérieux pour accélérer les simulations de mécanique des fluides turbulents. Dans l'industrie, cette robustesse est exploitée pour la prévision probabiliste avec DeepAR [76], utilisé par Amazon pour sa chaîne logistique. Ce modèle apprend une distribution de probabilité future à chaque pas de temps, permettant de quantifier l'incertitude via des simulations de Monte Carlo. Enfin, la capacité "génération constructive" des RNN a été pionnière dans la création artistique. Le modèle Performance RNN [77], développé par Google Magenta, a montré qu'un LSTM pouvait générer des performances de piano expressives (avec nuances de vitesse et de timing) en traitant la musique non pas comme une partition rigide, mais comme une séquence temporelle continue d'événements, prouvant que les RNN peuvent capturer des structures hiérarchiques globales (phrasé musical) tout en gérant des détails micro-temporels.

5.3.4 . Transformer

Si les réseaux récurrents ont introduit la mémoire et les réseaux convolutionnels la localité, l'architecture Transformer a proposé un changement de paradigme radical en postulant que l'interaction entre les éléments d'une séquence doit être modélisée par une relation directe de contenu à contenu, et non par une contrainte de proximité spatiale ou temporelle. Cette architecture, devenue l'épine dorsale de l'IA générative moderne, repose sur le mécanisme d'attention.

Histoire : De l'alignement au "Pointer Network" et à l'Attention pure

L'émergence du Transformer est le fruit d'une lente maturation visant à résoudre le goulot d'étranglement des architectures Encodeur-Décodeur récurrentes (RNN) REF. Dans le paradigme Seq2Seq classique [72], toute l'information de la phrase source devait être compressée dans un unique vecteur de contexte de taille fixe, entraînant une perte d'information critique sur les longues séquences. Une première solution [78] introduit un mécanisme d'attention additive permettant au décodeur de "chercher" (search) et d'aligner (align) les parties pertinentes de la phrase source à chaque étape de la génération. Ici, l'attention n'était encore qu'un module auxiliaire greffé sur des RNN. Une seconde étape conceptuelle fut franchie avec les Pointer Networks [79] où le réseau de neurones peut apprendre à résoudre des problèmes combinatoires (comme l'enveloppe convexe) en utilisant l'attention comme un pointeur pour sélectionner des éléments de l'entrée comme sortie, plutôt que de générer des symboles abstraits. Cela a ancré l'idée que le mécanisme de sélection basé sur le contenu ("Content-based addressing") était suffisamment puissant pour structurer la sortie. La rupture définitive survient avec l'article At-

tention Is All You Need [38]. Les auteurs ont démontré que la récurrence, jugée jusqu'alors indispensable pour encoder l'ordre séquentiel, était en réalité superflue et limitante pour la parallélisation. En ne conservant que le mécanisme d'attention (devenu Self-Attention), ils ont permis un traitement parallèle massif des séquences, réduisant la distance de propagation de l'information entre deux mots quelconques à une constante $O(1)$, contre $O(N)$ pour un RNN.

Mécanisme d'interaction et complexité

Contrairement aux architectures précédentes qui traitent la séquence par voisinage spatial ou récursivité temporelle, le Transformer repose sur un mécanisme d'interaction directe et globale : l'attention. Ce processus permet à chaque élément de la séquence de construire sa propre représentation en agrégeant l'information de tous les autres éléments, pondérée par leur pertinence contextuelle. Cette interaction est formalisée par le mécanisme "Query-Key-Value". Pour chaque élément d'entrée x_i , trois vecteurs sont générés par projection linéaire via des matrices de poids apprenables W^Q, W^K, W^V : une Requête $q_i = x_i W^Q$, une Clé $k_i = x_i W^K$ et une Valeur $v_i = x_i W^V$.

Le cœur du calcul réside dans la mesure de compatibilité entre ces vecteurs, appelé score d'attention. Pour construire une nouvelle représentation d'un élément x_i dans son contexte (c'est-à-dire le reste de la séquence), la requête associée q_i est comparée aux clés de chaque élément de la séquence $(k_j)_j$, formant une séquence de score d'attention $(a_{i,j})_j$. Cette comparaison s'effectue souvent par un produit scalaire, mis à l'échelle par la racine de la dimension des clés d_{att} pour stabiliser les gradients. Le score d'attention brute $a_{i,j}$ est généralement donné par :

$$a_{i,j} = \frac{\langle q_i, k_j \rangle}{\sqrt{d_{att}}}$$

Ces scores bruts sont ensuite convertis en une distribution de probabilité $(p_{i,j})_j$ par l'application d'une fonction Softmax :

$$p_{i,j} = \text{softmax}((a_{i,k})_k)_j = \frac{\exp(a_{i,j})}{\sum_k \exp(a_{i,k})}$$

La nouvelle représentation de x_i , notée y_i , est calculée comme une somme des vecteurs de Valeur $(v_j)_j$ pondérés par les poids d'attention $(p_{i,j})_j$:

$$y_i = \sum_j p_{i,j} v_j$$

Il est intéressant de noter que l'opération d'attention est invariante par permutation (elle traite la séquence comme un ensemble, un "sac de mots"). Il est donc nécessaire d'injecter explicitement d'informations de position (Positional Encodings) a priori dans la séquence $(x_i)_i$ pour reconstruire la topologie temporelle ou spatiale de la donnée si elle n'y est pas présent initialement.

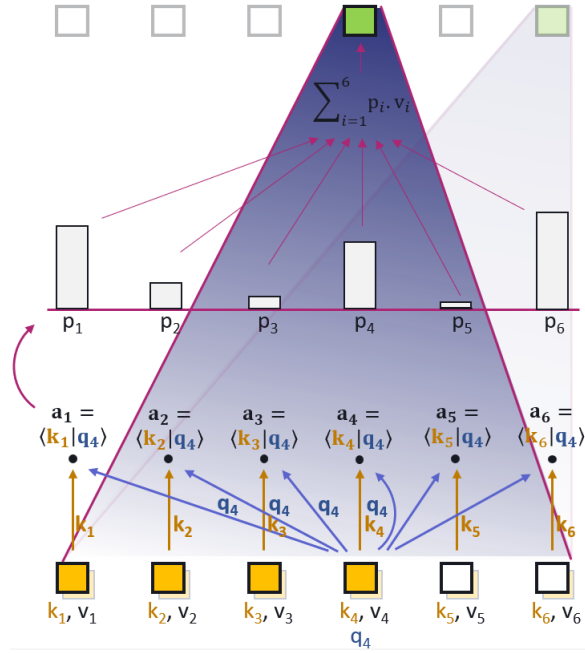


Figure 5.5 – Illustration du mécanisme d'Auto-Attention par produit scalaire

L'illustration 5.14 détaille le processus du calcul de la représentation contextuelle d'un élément cible x_4 (représenté en vert). Comme expliqué à l'instant, les éléments de la séquence sont projetés dans l'espace des requêtes, clés et valeurs, les scores d'attention sont calculés puis la pondération d'attention en est déduite et la nouvelle représentation de x_4 que nous notons y_4 , est calculée en effectuant une somme des valeurs pondérées par les poids d'attention :

$$\begin{aligned}
 q_4 &= x_4 W^Q, & k_j &= x_j W^K, & v_i &= x_i W^V \\
 a_{4,j} &= \frac{\langle q_4, k_j \rangle}{\sqrt{d_{att}}}, & p_{4,j} &= \frac{\exp(a_{4,j})}{\sum_k \exp(a_{4,k})}, & y_4 &= \sum_j p_{4,j} v_j
 \end{aligned}$$

Ainsi, le vecteur résultant y_4 est une synthèse dynamique du contenu de la séquence.

Le mécanisme canonique d'auto-attention se décline en deux variantes fondamentales pour répondre à des contraintes structurelles spécifiques : le respect de la causalité temporelle et l'intégration d'informations exogènes. La première variante, l'Attention Masquée (Masked Self-Attention), est indispensable aux tâches de génération séquentielle ou de simulation, où la prédiction de l'état présent ne doit physiquement pas dépendre du futur. Cette causalité est induite par une modification de la matrice des scores d'attention avant l'étape de normalisation : en forçant vers $-\infty$ les scores associés aux indices futurs, on garantit que la fonction Softmax leur attribue une probabilité strictement nulle. La partie gauche de la figure 5.15 illustre ce mécanisme : lors du calcul de la représentation pour la position 4 (carré vert), l'accès aux positions ultérieures 5 et 6 est bloqué par le masque. Le vecteur de sortie y_4 est ainsi construit exclusive-

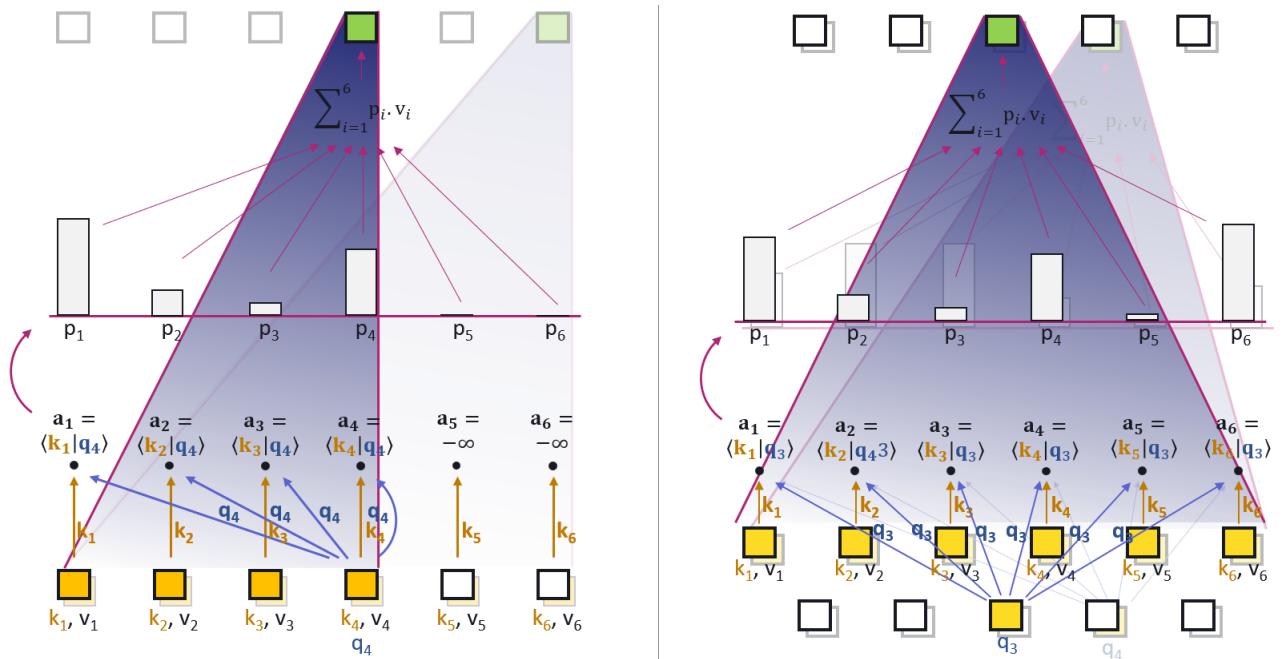


Figure 5.6 – Adaptations du mécanisme d'attention : restriction causale (gauche) et interaction inter-séquences (droite)

ment par l'agrégation des valeurs passées et présentes (v_1 à v_4), préservant l'intégrité causale du flux de données.

La seconde variante, l'Attention Croisée (Cross-Attention), permet le transfert d'information entre deux séquences distinctes, une opération centrale pour les tâches de traduction ou de reconstruction conditionnelle. Cette architecture repose sur une distribution asymétrique des rôles : la séquence source (qui détient l'information) projette les Clés (K) et les Valeurs (V), tandis que la séquence cible (qui cherche à s'enrichir ou se construire) émet les Requêtes (Q). La partie droite de la figure 5.15 détaille cette interaction : la séquence du bas représente le flux cible, dont le troisième élément émet une requête q_3 . Celle-ci est comparée à l'ensemble des clés k issues de la séquence source (au milieu), permettant de pondérer les valeurs v_1 à v_6 correspondantes. Le vecteur résultant est donc une injection dynamique du contexte source pertinent au sein de la trajectoire cible, pilotée par les besoins de cette dernière.

L'expressivité du Transformer repose sur la parallélisation du mécanisme d'attention unitaire et son intégration dans différents blocs. Pour capturer des relations de natures variées (syntaxiques, sémantiques, ou causales par exemple) à différentes échelles, le modèle utilise l'Attention Multi-Têtes (Multi-Head Attention). Au lieu de calculer une unique matrice d'attention sur la dimension totale du modèle d_{model} , l'entrée est projetée linéairement h fois dans des sous-espaces de dimension réduite $d_{att} = d_{model}/h$. Chaque "tête" calcule sa propre attention indépendamment, permettant au réseau de se focaliser simultanément sur différents aspects

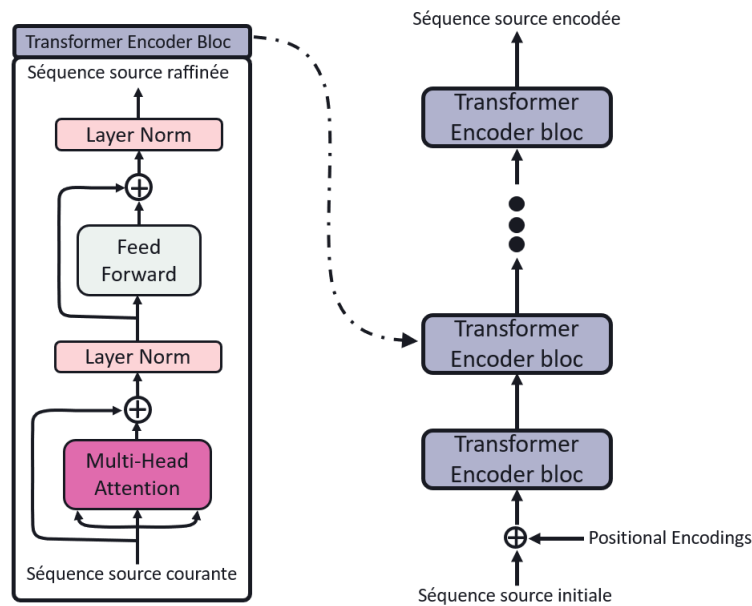


Figure 5.7 – Architecture du bloc Encodeur du Transformer

de la séquence. Les sorties de ces h têtes sont ensuite concaténées et reprojétées par une matrice linéaire finale W^O pour restaurer la dimension originale. Mathématiquement, cela permet de recombinaison les informations extraites de chaque sous-espace pour former une représentation unifiée.

La illustration 5.12 présente l'architecture de l'Encodeur, dédiée à l'analyse et à la construction d'une représentation contextuelle robuste de la séquence d'entrée. Elle est constituée d'un empilement de N blocs identiques. Chaque bloc s'articule autour de deux sous-modules fonctionnels : l'Attention Multi-Têtes, qui capture les interactions globales entre les différentes positions, et un réseau de neurones dense (Feed-Forward Network - FFN) appliqué indépendamment à chaque position pour traiter les caractéristiques. Pour permettre de la profondeur au réseau, chaque sous-module est systématiquement encapsulé par une connexion résiduelle - qui additionne l'entrée du module à sa sortie pour préserver le flux de gradient - suivie d'une normalisation de couche (Layer Norm) assurant stabilité et une bonne propagation du gradient dans toutes les couches. Enfin, l'architecture étant invariante par permutation, l'ajout dès l'entrée d'un Encodage Positionnel est indispensable pour injecter la topologie temporelle ou spatiale dans les représentations vectorielles.

L'illustration 5.13 détaille l'architecture du Décodeur, conçue pour la génération autorégressive. Bien qu'elle hérite de la structure modulaire stratifiée de l'encodeur, elle s'en distingue par l'intégration dans chaque bloc de mécanismes spécifiques dédiés à la prédiction. Le premier module est une Attention Multi-Têtes Masquée (Masked Self-Attention). Ce composant

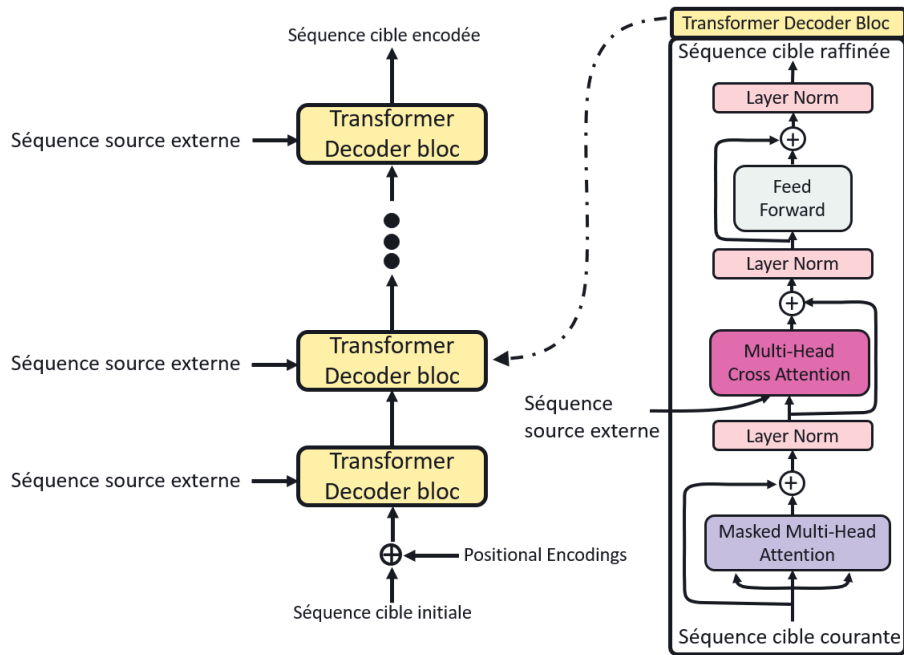


Figure 5.8 – Architecture du bloc Décodeur de Transformer

permet aux éléments de la séquence cible d'assimiler exclusivement les informations des états antérieurs, garantissant le respect de la causalité temporelle nécessaire à la génération. Le décodeur se singularise ensuite par l'insertion d'un troisième sous-module, intercalé avant le FFN : l'Attention Croisée (Cross-Attention). Ce module est l'interface de conditionnement du modèle ; il incorpore à la séquence cible (qui fournit les requêtes Q) l'information contextuelle extraite d'une source externe (l'encodeur, qui fournit les clés K et les valeurs V). Chacun de ces trois sous-modules (Masked Self-Attention, Cross-Attention, FFN) est soumis au même schéma de stabilisation par connexion résiduelle et normalisation que l'encodeur, assurant une cohérence dynamique à travers tout le réseau.

Dans une configuration complète de type "Séquence vers Séquence" (Seq2Seq), telle que celle utilisée pour la traduction automatique ou la simulation physique, la sortie de la pile d'encodeurs est connectée à l'entrée source de la pile de décodeurs. Cette architecture bipartite, illustrée figure 5.11, permet de transformer une séquence d'entrée complexe (scénario tactique, signal bruité) en une représentation latente continue. À partir de cet espace, le décodeur reconstruit pas à pas la séquence de sortie cible (signal reconstruit), agissant ainsi comme un traducteur universel capable de modéliser des fonctions de transfert hautement non-linéaires entre des signaux de natures variées.

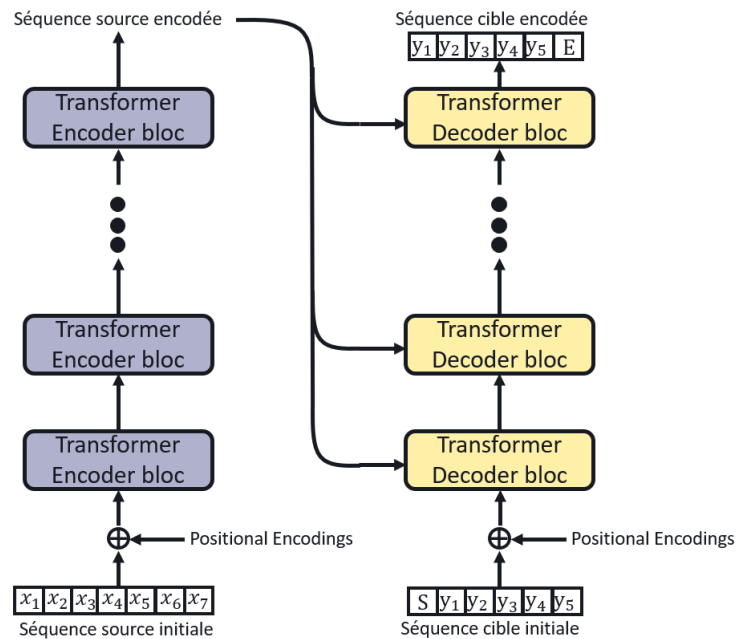


Figure 5.9 – Architecture Transformer complète pour le paradigme Séquence-vers-Séquence

Le Transformer dans le texte : La divergence des architectures

Dans le traitement du langage naturel (NLP), le Transformer a provoqué une véritable explosion cambrienne des modèles, se scindant en trois familles distinctes. La première, celle des Encodeurs, est incarnée par BERT [80]. Utilisant une attention bidirectionnelle, ces modèles excellent dans la compréhension et la classification, car chaque mot a accès au contexte passé et futur simultanément. La seconde, celle des Décodeurs, est dominée par la lignée GPT [39], [81]. Ici, l'attention est causale (masquée vers le futur), optimisée pour la génération autorégressive. C'est cette branche qui a mis en évidence les "lois d'échelle" (Scaling Laws), montrant que la performance de prédiction du prochain token suit une loi de puissance en fonction du nombre de paramètres et de données, ouvrant la voie aux gros modèles de langage (Large Language Model - LLM) actuels. La troisième famille, Encodeur-Décodeur, reste fidèle à l'architecture originale [38] pour les tâches de traduction ou de résumé. Le modèle T5 [82] a poussé ce paradigme à son extrême en reformulant toute tâche NLP (y compris la classification) comme un problème de génération de texte-vers-texte.

Le Transformer dans les systèmes temporels : Promesses et controverses

L'application des Transformers aux séries temporelles continues (consommation énergétique, trafic, météo) a fait l'objet de recherches intenses [83]. L'attrait principal réside dans la capacité théorique de l'attention à capturer des corrélations à très long terme et des saisonnalités complexes que les RNN peinent à retenir. Des architectures spécifiques ont été proposées

pour briser la complexité quadratique. Informer [84] introduit une attention "ProbSparse" pour sélectionner uniquement les interactions dominantes, réduisant la complexité à $O(N \log N)$. Autoformer [85] va plus loin en remplaçant le produit scalaire par une auto-corrélation pour mieux capturer les périodicités. Cependant, l'efficacité réelle des Transformers sur des signaux continus est contestée. Une étude [86] assure qu'un simple modèle linéaire bien calibré (DLi-near) surpassait souvent des Transformers complexes sur les benchmarks standards. La raison invoquée est que l'attention, conçue pour la sémantique discrète, tend à sur-interpréter le bruit dans les signaux continus et perd l'information d'ordre temporel cruciale, malgré les encodages positionnels. Néanmoins, des approches récentes comme PatchTST [87], qui segmentent le signal en patches (comme en vision) avant d'appliquer l'attention, semblent redonner l'avantage aux Transformers en traitant des dynamiques locales plutôt que des points isolés.

Le Transformer dans l'image : Patches et hiérarchie

L'hégémonie des CNN en vision a été remise en cause par le Vision Transformer (ViT) [41]. En découpant l'image en une séquence de patches carrés traités comme des mots, ViT a prouvé qu'un Transformer pur, sans biais inductif de convolution, pouvait atteindre l'état de l'art, à condition d'être pré-entraîné sur des volumes de données massifs (JFT-300M, [88]). Pour pallier le coût quadratique sur les images haute résolution et le manque de localité, l'architecture Swin Transformer [89] a réintroduit une structure hiérarchique similaire aux CNN. En calculant l'attention uniquement à l'intérieur de fenêtres locales glissantes (Shifted Windows), Swin combine la modélisation globale des Transformers avec l'efficacité locale des convolutions, devenant le standard actuel pour la segmentation et la détection d'objets.

Généralisation : Physique et Prise de décision

La capacité du Transformer à modéliser des graphes d'interaction arbitraires en fait un outil puissant pour la physique et la biologie. L'exemple le plus spectaculaire est AlphaFold 2 [90], qui a résolu le problème du repliement des protéines. Son module central, l'Evoformer, est une variante du Transformer qui traite la protéine comme un graphe dynamique, mettant à jour itérativement la représentation de la séquence d'acides aminés et la matrice de distances 3D par des mécanismes d'attention triangulaire. Enfin, dans le domaine du contrôle et de la simulation, le Decision Transformer [91] a reformulé l'apprentissage par renforcement comme un problème de modélisation de séquence. Au lieu d'estimer des fonctions de valeur ou des gradients de politique, ce modèle prédit simplement l'action suivante conditionnée par les états passés et la récompense désirée (le "Return-to-go"). Cette approche "généraliste" du contrôle permet d'appliquer les techniques de pré-entraînement des LLM à la robotique ou à la navigation d'agents autonomes, unifiant ainsi perception, prédiction physique et prise de décision sous une même architecture.

5.3.5 . L'architecture Transformer

Si les réseaux récurrents ont introduit la mémoire et les réseaux convolutionnels la localité, l'architecture Transformer a proposé un changement de paradigme radical en postulant que l'interaction entre les éléments d'une séquence doit être modélisée par une relation directe de contenu à contenu, et non par une contrainte de proximité spatiale ou temporelle. Cette architecture, devenue l'épine dorsale de l'IA générative moderne, repose sur le mécanisme d'attention.

Genèse : Du goulot d'étranglement récurrent à l'Attention pure

L'émergence du Transformer est le fruit d'une lente maturation visant à résoudre le goulot d'étranglement des architectures Encodeur-Décodeur récurrentes (RNN). Dans le paradigme Seq2Seq classique [72], toute l'information de la phrase source devait être compressée dans un unique vecteur de contexte de taille fixe, entraînant une perte d'information critique sur les longues séquences. Une première solution [78] introduit un mécanisme d'attention additive permettant au décodeur de "chercher" (search) et d'aligner (align) les parties pertinentes de la phrase source à chaque étape de la génération. Ici, l'attention n'était encore qu'un module auxiliaire greffé sur des RNN. Une seconde étape conceptuelle fut franchie avec les Pointer Networks [79] où le réseau de neurones apprend à résoudre des problèmes combinatoires en utilisant l'attention comme un pointeur pour sélectionner des éléments de l'entrée comme sortie. Cela a ancré l'idée que le mécanisme de sélection basé sur le contenu ("Content-based addressing") était suffisamment puissant pour structurer la sortie. La rupture définitive survient avec l'article Attention Is All You Need [38]. Les auteurs ont démontré que la récurrence, jugée jusqu'alors indispensable pour encoder l'ordre séquentiel, était en réalité superflue et limitante pour la parallélisation. En ne conservant que le mécanisme d'attention, ils ont permis un traitement parallèle massif des séquences, réduisant la distance de propagation de l'information entre deux mots quelconques à une constante $O(1)$, contre $O(N)$ pour un RNN.

Contrairement aux CNN ou RNN qui se définissent par une opération élémentaire, le Transformer se définit d'abord comme une architecture systémique modulaire. Nous privilégions ici une description du flux global d'information au sein de l'architecture avant d'en détailler les opérations mathématiques constitutives.

Architecture Macroscopique : Le paradigme Encodeur-Décodeur

Dans la configuration complète telle que proposée pour la traduction automatique [38], l'architecture adopte une structure bipartite illustrée par la figure 5.10, avec un encodeur à gauche et un décodeur à droite. L'objectif est de transformer une séquence source $(x_i)_i$ en une séquence cible $(y_i)_i$. La raison d'être de cette architecture réside d'une part dans l'exploitation du mécanisme d'attention pour augmenter la représentativité des séquences encodées, et d'autre part dans la parallélisation de l'entraînement. Ce second point permet la prédiction de la séquence $(y_i)_i$ en une seule passe, rompant avec la contrainte de séquentialité inhérente aux architectures récurrentes (RNN) qui imposait un traitement itératif pas à pas.

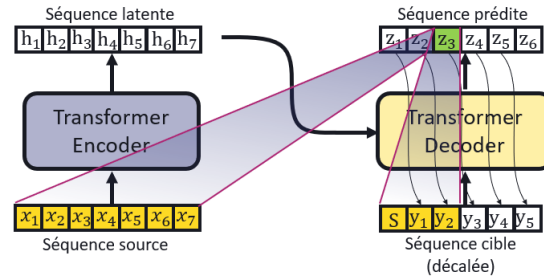


Figure 5.10 – Flux d'information macroscopique dans l'architecture Transformer Encodeur-Décodeur

La séquence source est traitée par l'encodeur qui, grâce au mécanisme d'auto-attention, enrichit la représentation de chaque élément en agrégeant l'information issue de l'ensemble du contexte (les autres éléments). Ce processus produit une séquence latente $(h_i)_i$ à haute teneur sémantique. Le décodeur exploite ensuite cette représentation pour générer la séquence de sortie.

Durant l'entraînement, le décodeur a accès à la vérité terrain complète $(y_i)_i$, condition nécessaire à cette parallélisation du calcul. Pour empêcher qu'il ne "triche" en accédant à la réponse y_k présente en entrée pour la prédire en sortie, on applique un décalage des entrées via un marqueur de début de séquence S . La séquence cible fournie au réseau est alors (S, y_1, \dots, y_N) . Ce décalage aligne mécaniquement l'information passée y_{k-1} face à l'objectif de prédiction présent z_k , exploité pour déterminer y_k . Associé à une interaction contrainte au sein de la séquence causale, ce dispositif assure que les prédictions sont effectuées dans les mêmes conditions de visibilité qu'en inférence, sans accès frauduleux à un avenir non encore prédit. Ce principe est illustré par le cône de perception violet de la représentation z_3 (en vert), utilisée pour prédire y_3 . On observe que z_3 agrège l'information complète de la source $(x_i)_i$ via l'attention croisée, mais n'accède côté cible qu'aux informations antérieures S, y_1 et y_2 .

Ainsi, à chaque étape, l'élément en cours de traitement accède aux informations des éléments antérieurs via une auto-attention masquée (garantissant la causalité), tout en intégrant le contexte global issu de la séquence latente $(h_i)_i$ grâce à un mécanisme d'attention croisée. Ce processus génère une séquence de représentations prédites $(z_i)_i$. Chaque vecteur z_i est finalement projeté sur le dictionnaire du vocabulaire pour produire une distribution de probabilité et prédire le prochain symbole. Ces prédictions sont réinjectées itérativement en entrée du décodeur (séquence cible décalée) pour poursuivre la génération.

Pour comprendre le fonctionnement interne, il faut disséquer ces deux piles. L'illustration 5.12 présente l'architecture de l'Encodeur, dédiée à l'analyse et à la construction d'une représentation contextuelle robuste de la séquence d'entrée. Elle est constituée d'un empilement de N blocs identiques. Chaque bloc s'articule autour de deux sous-modules fonctionnels : l'Attention Multi-Têtes, qui capture les interactions globales entre les différentes positions, et un réseau de neurones dense (Feed-Forward Network - FFN) appliqué indépendamment à chaque

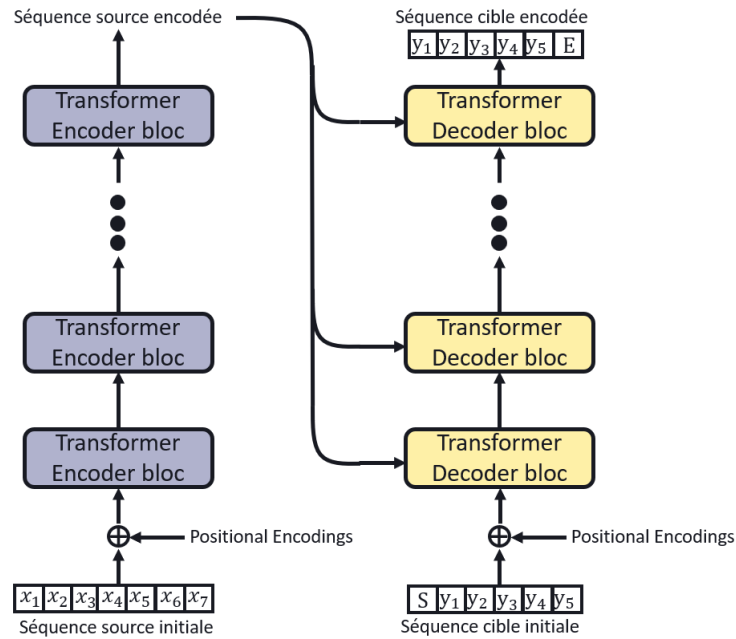


Figure 5.11 – Architecture Transformer complète pour le paradigme Séquence-vers-Séquence

position pour traiter les caractéristiques. Pour permettre de la profondeur au réseau, chaque sous-module est systématiquement encapsulé par une connexion résiduelle - qui additionne l'entrée du module à sa sortie pour préserver le flux de gradient - suivie d'une normalisation de couche (Layer Norm) assurant stabilité et une bonne propagation du gradient dans toutes les couches. Enfin, l'architecture étant invariante par permutation, l'ajout dès l'entrée d'un Encodage Positionnel est indispensable pour injecter la topologie temporelle ou spatiale dans les représentations vectorielles. L'illustration 5.13 détaille l'architecture du Décodeur, conçue pour la génération autorégressive. Bien qu'elle hérite de la structure modulaire stratifiée de l'encodeur, elle s'en distingue par l'intégration dans chaque bloc de mécanismes spécifiques dédiés à la prédiction. Le premier module est une Attention Multi-Têtes Masquée (Masked Self-Attention). Ce composant permet aux éléments de la séquence cible d'assimiler exclusivement les informations des états antérieurs, garantissant le respect de la causalité temporelle nécessaire à la génération. Le décodeur se singularise ensuite par l'insertion d'un troisième sous-module, intercalé avant le FFN : l'Attention Croisée (Cross-Attention). Ce module est l'interface de conditionnement du modèle; il incorpore à la séquence cible (qui fournit les requêtes Q) l'information contextuelle extraite d'une source externe (l'encodeur, qui fournit les clés K et les valeurs V). Chacun de ces trois sous-modules (Masked Self-Attention, Cross-Attention, FFN) est soumis au même schéma de stabilisation par connexion résiduelle et normalisation que l'encodeur, assurant une cohérence dynamique à travers tout le réseau.

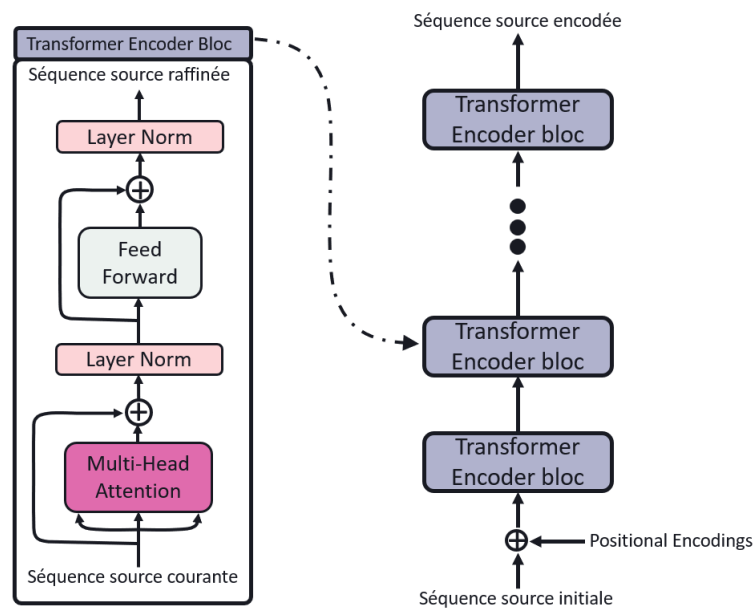


Figure 5.12 – Architecture du bloc Encodeur du Transformer

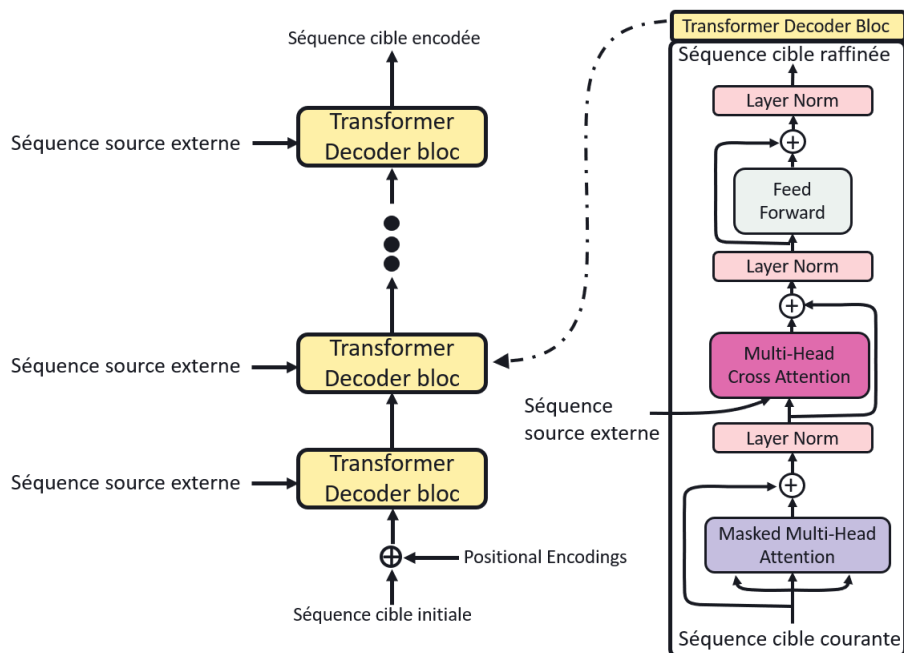


Figure 5.13 – Architecture du bloc Décodeur de Transformer

Micro-architecture : Le mécanisme d'Attention et ses variantes

Au cœur de ces blocs macroscopiques réside le véritable moteur de l'interaction : le mécanisme d'Attention. Contrairement aux architectures précédentes qui traitent la séquence par voisinage spatial ou récursivité temporelle, le Transformer repose sur un mécanisme d'interaction directe et globale. Ce processus permet à chaque élément de la séquence de construire sa propre représentation en agrégeant l'information de tous les autres éléments, pondérée par leur pertinence contextuelle. Cette interaction est formalisée par le mécanisme "Query-Key-Value". Pour chaque élément d'entrée x_i , trois vecteurs sont générés par projection linéaire via des matrices de poids apprenables W^Q, W^K, W^V : une Requête $q_i = x_i W^Q$, une Clé $k_i = x_i W^K$ et une Valeur $v_i = x_i W^V$. Le cœur du calcul réside dans la mesure de compatibilité entre ces vecteurs, appelé score d'attention. Pour construire une nouvelle représentation d'un élément x_i dans son contexte, la requête associée q_i est comparée aux clés de chaque élément de la séquence $(k_j)_j$. Cette comparaison s'effectue souvent par un produit scalaire, mis à l'échelle par la racine de la dimension des clés d_{att} pour stabiliser les gradients. Le score d'attention brut $a_{i,j}$ est généralement donné par :

$$a_{i,j} = \frac{\langle q_i, k_j \rangle}{\sqrt{d_{att}}}$$

Ces scores bruts sont ensuite convertis en une distribution de probabilité $(p_{i,j})_j$ par l'application d'une fonction Softmax :

$$p_{i,j} = \text{softmax}((a_{i,k})_k)_j = \frac{\exp(a_{i,j})}{\sum_k \exp(a_{i,k})}$$

La nouvelle représentation de x_i , notée y_i , est calculée comme une somme des vecteurs de Valeur $(v_j)_j$ pondérés par les poids d'attention $(p_{i,j})_j$:

$$y_i = \sum_j p_{i,j} v_j$$

L'illustration 5.14 détaille ce processus de calcul pour un élément cible x_4 (représenté en vert). Comme expliqué à l'instant, les éléments de la séquence sont projetés dans l'espace des requêtes, clés et valeurs. Les scores d'attention sont calculés, puis la pondération d'attention en est déduite. La nouvelle représentation y_4 est calculée en effectuant une somme des valeurs pondérées :

$$q_4 = x_4 W^Q, \quad k_j = x_j W^K, \quad v_i = x_i W^V \quad a_{4,j} = \frac{\langle q_4, k_j \rangle}{\sqrt{d_{att}}}, p_{4,j} = \frac{\exp(a_{4,j})}{\sum_k \exp(a_{4,k})}, y_4 = \sum_j p_{4,j} v_j$$

Ainsi, le vecteur résultant y_4 est une synthèse dynamique du contenu de la séquence. Ce mécanisme canonique se décline en deux variantes fondamentales pour répondre aux contraintes structurelles identifiées dans l'architecture macroscopique : le respect de la causalité temporelle (pour le décodeur) et l'intégration d'informations exogènes (pour l'interface). La première variante, l'Attention Masquée (Masked Self-Attention), est indispensable aux tâches de génération séquentielle, où la prédiction de l'état présent ne doit physiquement pas dépendre du

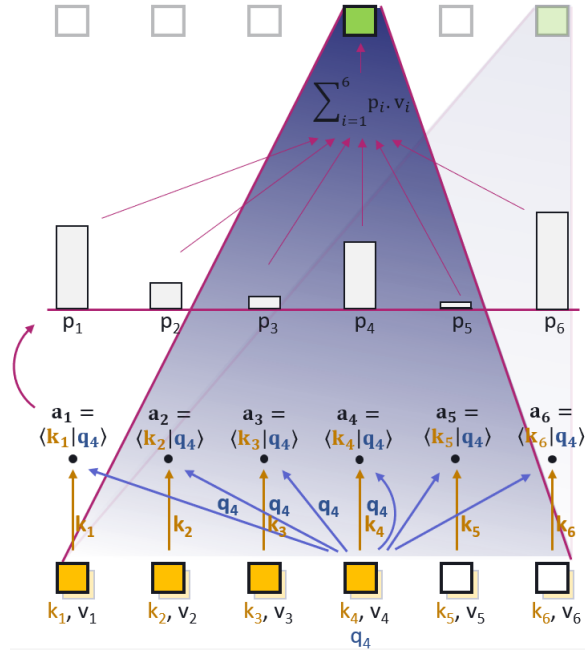


Figure 5.14 – Illustration du mécanisme d'Auto-Attention par produit scalaire

futur. Cette causalité est induite par une modification de la matrice des scores d'attention avant l'étape de normalisation : en forçant vers $-\infty$ les scores associés aux indices futurs, on garantit que la fonction Softmax leur attribue une probabilité strictement nulle. La partie gauche de la figure 5.15 illustre ce mécanisme : lors du calcul de la représentation pour la position 4 (carré vert), l'accès aux positions ultérieures 5 et 6 est bloqué par le masque. Le vecteur de sortie y_4 est ainsi construit exclusivement par l'agrégation des valeurs passées et présentes (v_1 à v_4), préservant l'intégrité causale du flux de données. La seconde variante, l'Attention Croisée (Cross-Attention), permet le transfert d'information entre deux séquences distinctes. Cette architecture repose sur une distribution asymétrique des rôles : la séquence source (l'encodeur) projette les Clés (K) et les Valeurs (V), tandis que la séquence cible (le décodeur) émet les Requêtes (Q). La partie droite de la figure 5.15 détaille cette interaction : la séquence du bas représente le flux cible, dont le troisième élément émet une requête q_3 . Celle-ci est comparée à l'ensemble des clés k issues de la séquence source (au milieu), permettant de pondérer les valeurs v_1 à v_6 correspondantes. Le vecteur résultant est donc une injection dynamique du contexte source pertinent au sein de la trajectoire cible. Enfin, l'expressivité du Transformer repose sur la parallélisation de ce mécanisme. Pour capturer des relations de natures variées (syntaxiques, sémantiques, ou causales) simultanément, le modèle utilise l'Attention Multi-Têtes (Multi-Head Attention). Au lieu de calculer une unique matrice d'attention, l'entrée est projetée linéairement h fois dans des sous-espaces de dimension réduite. Les sorties de ces h têtes sont ensuite concaténées et reprojetées par une matrice linéaire finale W^O .

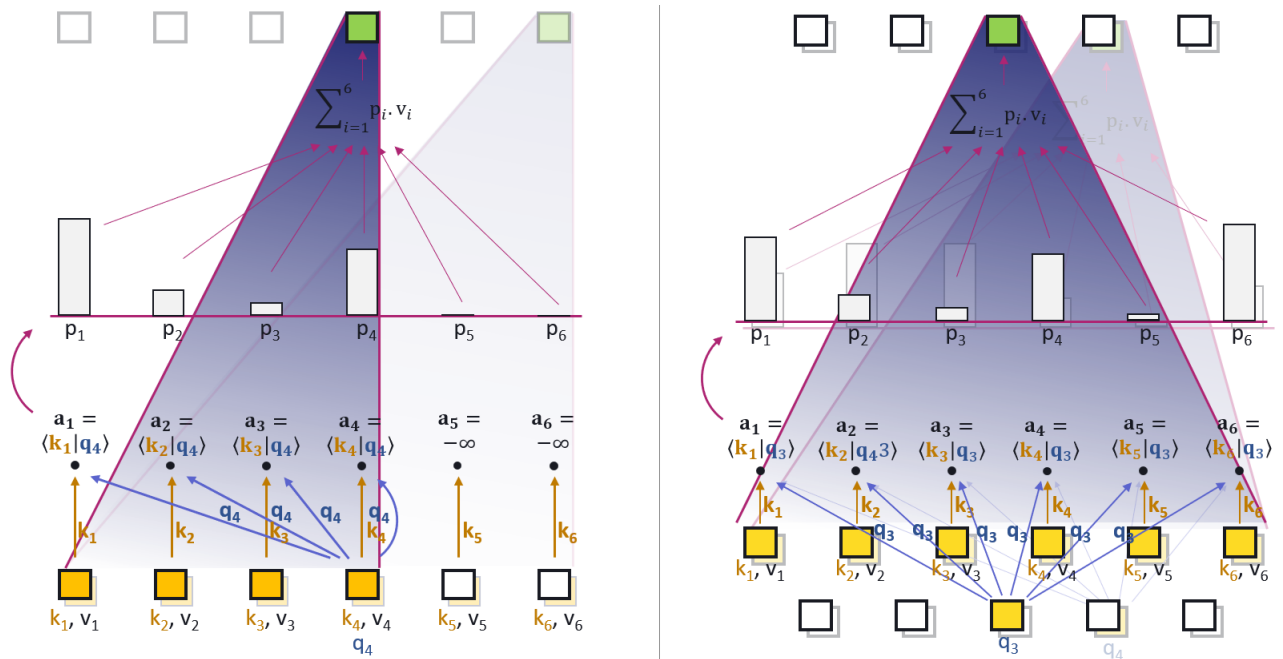


Figure 5.15 – Adaptations du mécanisme d'attention : restriction causale (gauche) et interaction inter-séquences (droite)

Le Transformer dans le texte : La divergence des architectures

Dans le traitement du langage naturel (NLP), le Transformer a provoqué une véritable explosion cambrienne des modèles, se scindant en trois familles distinctes. La première, celle des Encodeurs, est incarnée par BERT [80]. Utilisant une attention bidirectionnelle, ces modèles excellent dans la compréhension et la classification. La seconde, celle des Décodeurs, est dominée par la lignée GPT [39], [81]. Ici, l'attention est causale (masquée vers le futur), optimisée pour la génération autorégressive. C'est cette branche qui a mis en évidence les "lois d'échelle" (Scaling Laws), montrant que la performance de prédiction du prochain token suit une loi de puissance en fonction du nombre de paramètres et de données. La troisième famille, Encodeur-Décodeur, reste fidèle à l'architecture originale [38] pour les tâches de traduction. Le modèle T5 [82] a poussé ce paradigme à son extrême en reformulant toute tâche NLP (y compris la classification) comme un problème de génération de texte-vers-texte.

Le Transformer dans les systèmes temporels : Promesses et controverses

L'application des Transformers aux séries temporelles continues a fait l'objet de recherches intenses [83]. L'attrait principal réside dans la capacité théorique de l'attention à capturer des corrélations à très long terme que les RNN peinent à retenir. Des architectures spécifiques ont été proposées pour briser la complexité quadratique. Informer [84] introduit une attention "ProbSparse" pour sélectionner uniquement les interactions dominantes ($O(N \log N)$). Auto-

former [85] va plus loin en remplaçant le produit scalaire par une auto-corrélation. Cependant, l'efficacité réelle des Transformers sur des signaux continus est contestée. Une étude [86] assure qu'un simple modèle linéaire bien calibré (DLinear) surpasse souvent des Transformers complexes, car l'attention tend à sur-interpréter le bruit dans les signaux continus. Néanmoins, des approches récentes comme PatchTST [87], qui segmentent le signal en patches avant d'appliquer l'attention, semblent redonner l'avantage aux Transformers en traitant des dynamiques locales plutôt que des points isolés.

Le Transformer dans l'image : Patches et hiérarchie

L'hégémonie des CNN en vision a été remise en cause par le Vision Transformer (ViT) [41]. En découpant l'image en une séquence de patches carrés traités comme des mots, ViT a prouvé qu'un Transformer pur pouvait atteindre l'état de l'art, à condition d'être pré-entraîné sur des volumes de données massifs. Pour pallier le coût quadratique sur les images haute résolution et le manque de localité, l'architecture Swin Transformer [89] a réintroduit une structure hiérarchique similaire aux CNN. En calculant l'attention uniquement à l'intérieur de fenêtres locales glissantes (Shifted Windows), Swin combine la modélisation globale des Transformers avec l'efficacité locale des convolutions.

Généralisation : Physique et Prise de décision

La capacité du Transformer à modéliser des graphes d'interaction arbitraires en fait un outil puissant pour la physique et la biologie. L'exemple le plus spectaculaire est AlphaFold 2 [90], qui a résolu le problème du repliement des protéines. Son module central, l'Evoformer, est une variante du Transformer qui traite la protéine comme un graphe dynamique, mettant à jour itérativement la représentation par des mécanismes d'attention triangulaire. Enfin, dans le domaine du contrôle, le Decision Transformer [91] a reformulé l'apprentissage par renforcement comme un problème de modélisation de séquence. Au lieu d'estimer des fonctions de valeur, ce modèle prédit l'action suivante conditionnée par les états passés et la récompense désirée (le "Return-to-go"), unifiant ainsi perception, prédiction physique et prise de décision sous une même architecture.

5.3.6 . Ancrage dans la problématique

L'exploration des architectures de traitement de séquence met en lumière un éventail de mécanismes complémentaires pour la modélisation de notre simulateur de capteur, dont la pertinence doit être pondérée par les contraintes spécifiques des signaux radar. Les réseaux convolutionnels, par leur biais inductif de localité, offrent une approche adaptée pour modéliser les interactions à courte portée, telles que les interférences immédiates entre impulsions proches au sein d'un même train. Cependant, leur architecture à fenêtre glissante impose une limitation structurelle majeure : la difficulté à maintenir un état mémoire persistant sur des horizons temporels arbitrairement longs, ce qui peut s'avérer insuffisant pour reproduire fidèlement les processus de pistage temporel qui nécessitent de lier des événements très distants.

De leur côté, les architectures récurrentes (RNN) et les modèles d'espaces d'états (SSM) présentent une affinité naturelle avec la causalité physique du capteur, mimant le comportement des algorithmes de traitement du signal qui mettent à jour des pistes au gré des réceptions. Néanmoins, leur usage impliquerait un changement de paradigme par rapport à notre approche orientée "traduction". Ces modèles excellent dans le traitement séquentiel flux à flux, mais leur application à une tâche de transformation globale de séquence (Seq2Seq) sur de très longs scénarios est complexe. Le goulot d'étranglement du vecteur de contexte, censé compresser toute l'information de la séquence d'entrée avant la génération, devient rapidement prohibitif face à la densité des données radar, limitant leur capacité à reconstruire fidèlement l'ensemble du scénario en une seule passe.

Enfin, l'architecture Transformer et le mécanisme d'attention apportent une capacité de modélisation contextuelle globale, permettant à chaque impulsion d'interagir directement avec l'ensemble de la séquence. Cette propriété est puissante pour capturer des corrélations complexes non-locales et apprendre la fonction de transfert globale du simulateur sans les contraintes de compression des RNN. Toutefois, l'application de ce modèle exige une vigilance particulière quant à sa complexité quadratique, qui peut devenir prohibitive face à la haute densité des flux d'impulsions radar, ainsi qu'à la nécessité d'adapter l'encodage positionnel pour traiter le temps continu irrégulier des PDW plutôt que des indices discrets.

6 - AVERTISSEMENT

La composition de la page de couverture doit être respectée pour la diffusion de la thèse sur www.theses.fr et pour le dépôt légal de la thèse, qui est obligatoire pour l'obtention du diplôme (cf. articles 24 et 25 de l'arrêté du 25 mai 2016 fixant le cadre national de la formation et les modalités conduisant à la délivrance du diplôme national de doctorat).

Les consignes et les recommandations ci-après ont pour objet d'assurer une **homogénéité graphique** pour toutes les thèses soutenues à l'université Paris-Saclay et de les rendre **immédiatement reconnaissables**.

Elles ont également pour objet de donner un cadre de référence permettant d'éviter qu'un lecteur futur puisse avoir des **doutes sur la conformité de la thèse ou du jury**. L'université reçoit régulièrement des demandes d'informations, au sujet de thèses, pour lesquelles il y a des questionnements sur la conformité du jury ou bien des incohérences entre les informations qui figurent sur la couverture de la thèse, d'une part, et les méta-données de la thèse visibles sur www.theses.fr, d'autre part.

Il est rappelé que ces consignes et recommandations ne s'appliquent que pour le dépôt légal de la thèse et sa diffusion via le portail www.theses.fr. **Ce canal de diffusion n'est pas exclusif**. D'autres formats de page de couverture peuvent être librement utilisés par les auteurs sur d'autres canaux de diffusion (par exemple : pour afficher le nom et le logo d'une organisation qui aurait co-financé la thèse et pour la diffusion au sein de cette organisation), à condition que les informations requises pour la citation complète de la thèse de doctorat figurent. C'est-à-dire : au minimum : nom et prénom de l'auteur, titre de la thèse, date, lieu et établissement de soutenance (université Paris-Saclay et le cas échéant un établissement partenaire en cas de cotutelle internationale de thèse), ainsi que le logo de l'université Paris-Saclay et le cas échéant d'une université étrangère partenaire en cas de cotutelle internationale de thèse.

7 - COMPOSITION GÉNÉRALE, CHARTE GRAPHIQUE

7.1 . COMPOSITION DU DOCUMENT

Les deux premières pages sont consacrées aux informations institutionnelles.

Une troisième page peut être ajoutée pour compléter les informations institutionnelles réglementaires des deux premières pages. Par exemple, pour donner des informations sur l'organisme d'accueil ou financeur et afficher leurs logos, pour décrire brièvement un cadre partenarial, pour fournir les noms de personnalités invitées à siéger aux côtés du Jury pour la soutenance, pour afficher le logo du laboratoire etc.

La page des remerciements est alors placée en 3^e ou 4^e page, selon qu'une 3^e page a été ajoutée ou non pour apporter ces compléments d'informations.

7.2 . QUELS LOGOS FAIRE FIGURER ?

Il ne doit figurer sur la **page de couverture de thèse**, aucun autre logo que le **logo de l'université Paris-Saclay** et, en cas de cotutelle internationale de thèse, le logo de l'université partenaire étrangère qui délivre également le diplôme de doctorat pour cette thèse.

Il ne doit figurer sur la **seconde page**, aucun autre logo que le **logo de l'école doctorale**. Les logos institutionnels en vigueur de l'université Paris-Saclay et des écoles doctorales sont fournis au paragraphe 7.2.

Les autres logos, comme celui du laboratoire, d'une entreprise, d'une composante, d'un établissement-composante, d'une université membre associée, d'un organisme de recherche ou de toute autre organisation partenaire de la thèse, peuvent être regroupés dans une troisième page intérieure, avant la page des remerciements, mais ne doivent pas figurer pas sur les deux premières pages.

7.3 . POLICES DE CARACTÈRES ET COULEURS

Les polices de caractère à utiliser sont : Open Sans ou Segoe UI ou Tahoma ou Ebrima. Il ne faut utiliser qu'**une seule police de caractère**.

Sur les 3 premières pages, seules deux couleurs de police sont utilisées, noir et prune (R : 99 V : 0 B : 60). Dans le reste du document, vous pouvez utiliser d'autres couleurs de police, si nécessaire, en veillant à ce qu'elles appartiennent à la palette de couleurs de la charte graphique de l'université Paris-Saclay. D'autres nuances de couleurs peuvent être utilisées parmi

RVB 99 0 60	RVB 49 62 72	RVB 124 135 143	RVB 213 218 223
RVB 198 11 70	RVB 237 20 91	RVB 238 52 35	RVB 243 115 32
RVB 124 42 144	RVB 125 106 175	RVB 198 103 29	RVB 254 188 24
RVB 0 78 125	RVB 14 135 201	RVB 0 148 181	RVB 70 195 210
RVB 0 128 122	RVB 64 183 105	RVB 140 198 62	RVB 213 223 61

Figure 7.1 – Palette de couleurs de la charte graphique

les nuances de la palette de l'UPSaclay.

La [charte graphique de l'Université](#) peut être téléchargée sur l'intranet pour plus d'information.

Sur la couverture de la thèse, le **titre** est en police normale de taille 20, de couleur prune et la **traduction du titre** est en police normale de taille 12, de couleur noire et en italique. Si le titre et sa traduction sont très longs, la police peut éventuellement être réduite, mais sans descendre en dessous d'une police 14 pour le titre et d'une police 10 pour la traduction du titre.

8 - INFORMATIONS GÉNÉRALES SUR LA PAGE DE COUVERTURE

Les informations figurant sur la page de couverture de la thèse doivent être cohérentes avec le diplôme et avec les métadonnées de la thèse sur le portail nationale des thèses www.theses.fr.

8.1 . TITRE DE LA THÈSE ET LANGUE(S)

Le **titre de la thèse** doit être fourni en **français** et en **anglais**. Par défaut, le titre est en français et la traduction du titre est en anglais. Cependant, lorsque la thèse est rédigée en anglais, le titre peut être fourni en anglais et la traduction en français.

Les affiliations (université de rattachement...) peuvent, le cas échéant, être fournies en anglais pour des membres étrangers du Jury. La langue par défaut restant le français.

Tous les autres éléments de la couverture de la thèse sont en français, les noms des entités (école doctorale, unité de recherche, référent etc.) ainsi que les titres des membres du jury (Professeur, Maître de Conférences etc.). Les correspondances entre titres étrangers et français peuvent être trouvées sur le site du ministère ([GALAXIE](#))¹.

8.2 . SPÉCIALITÉ DE DOCTORAT

La spécialité de doctorat doit faire partie des spécialités pour lesquelles l'école doctorale est accréditée (en pratique : cela implique que vous devez pouvoir la sélectionner dans le menu déroulant des spécialités dans Adum).

La spécialité de doctorat retenue, via le menu déroulant dans Adum, sera celle qui figurera sur le diplôme.

Si votre spécialité n'apparaît pas, il faut contacter le directeur de votre école doctorale.

1. https://www.galaxie.enseignementsup-recherche.gouv.fr/ensup/pdf/EC_pays_etrangers/Tableau_comparaison_au_26_septembre_2012.pdf

8.3 . UNITÉ DE RECHERCHE

L'unité de recherche dans laquelle la thèse a été préparée est précisée sur la couverture de la thèse. Le nom de l'unité est cité en respectant les règles de signature officielles, telles qu'elles ont été convenues entre les tutelles des unités de recherche liées à l'université Paris-Saclay.

Pour les trouver : il faut sélectionner votre unité de recherche via la barre de sélection depuis cette page web : <https://www.universite-paris-saclay.fr/fr/signature> et copier-coller l'adresse de l'unité de recherche sur la couverture de thèse. Puis mettre l'acronyme officiel en premier et le nom des tutelles ensuite, entre parenthèses, dans l'ordre où elles sont indiquées sur <https://www.universite-paris-saclay.fr/fr/signature>. Par exemple, pour IJCLab :

- Voici ce qu'on récupère par un copié-collé depuis l'adresse ci-dessus : « *Université Paris-Saclay, CNRS, IJCLab, 91405, Orsay, France* ».
- Voici comment faire la citation sur la couverture de thèse : « *IJCLab (Université Paris-Saclay, CNRS)* ».

Si la thèse a été préparée dans deux unités de recherche (travaux interdisciplinaires, cotutelle internationale, mobilité...) merci de citer les deux unités de recherche.

Si vous êtes doctorant de l'université Paris-Saclay mais ne trouvez pas votre unité dans la liste, votre unité ne fait probablement pas partie de l'université. Dans ce cas, et à défaut d'une recommandation commune entre l'université et votre unité, complétez la ligne "unité de recherche" de votre page de titre en suivant les recommandations de votre unité de recherche.

La mention de l'université Paris-Saclay comme établissement de soutenance de votre thèse sera automatique en utilisant le modèle de page de couverture de l'université Paris-Saclay.

8.4 . LE RÉFÉRENT

Les référents sont à choisir, en cohérence avec ce qui figure dans votre dossier d'inscription, parmi les composantes, établissements-composantes et universités membres associés de l'Université Paris-Saclay :

- Faculté de droit, économie et gestion,
- Faculté de médecine
- Faculté de pharmacie
- Faculté des sciences d'Orsay
- Faculté des sciences du sport
- AgroParisTech
- Institut d'Optique
- ENS Paris-Saclay
- CentraleSupélec
- Université de Versailles-Saint-Quentin-en-Yvelines

- Université d'Évry Val d'Essonne
- École Nationale d'Architecture de Versailles

8.5 . GRADUATE SCHOOL

La Graduate School est à choisir en cohérence avec votre sujet de thèse et ce qui figure dans votre dossier d'inscription, parmi la ou les Graduate Schools de l'Université Paris-Saclay de rattachement de votre école doctorale ou de votre pôle d'école doctorale :

- Biosphera
- Chimie
- Informatique et sciences du numérique
- Droit
- Économie - Management
- Géosciences, climat, environnement et planètes
- Humanités et Sciences du Patrimoine
- Life Sciences and Health
- Mathématiques
- Physique
- Santé et médicaments
- Santé publique
- Sciences de l'ingénierie et des systèmes
- Sociologie et Science Politique
- Sport, mouvement et facteurs humains

8.6 . ÉCOLE DOCTORALE

- n°127 : astronomie et astrophysique d'Île-de-France (AAIF)
- n°129 : sciences de l'environnement d'Île-de-France (SEIF)
- n°564 : physique en Île-de-France (PIF)
- n°566 : sciences du sport, de la motricité et du mouvement humain (SSMMH)
- n°567 : sciences du végétal : du gène à l'écosystème (SEVE)
- n°568 : signalisations et réseaux intégratifs en biologie (Biosigne)
- n°569 : innovation thérapeutique : du fondamental à l'appliqué (ITFA)
- n°570 : santé publique (EDSP)
- n°571 : sciences chimiques : molécules, matériaux, instrumentation et biosystèmes (2MIB)
- n°572 : ondes et matière (EDOM)
- n°573 : interfaces : matériaux, systèmes, usages (INTERFACES)
- n°574 : mathématiques Hadamard (EDMH)
- n°575 : electrical, optical, bio : physics and engineering (EOBE)
- n°576 : particules hadrons énergie et noyau : instrumentation, imagerie, cosmos et simulation (PHENIICS)
- n°577 : structure et dynamique des systèmes vivants (SDSV)

- n°579 : sciences mécaniques et énergétiques, matériaux et géosciences (SMEMaG)
- n°580 : sciences et technologies de l'information et de la communication (STIC)
- n°581 : agriculture, alimentation, biologie, environnement, santé (ABIES)
- n°582 : cancérologie : biologie - médecine - santé (CBMS)
- n°629 : Sciences sociales et humanités (SSH)
- n°630 : Droit, Économie, Management (DEM)

8.7 . LIEU ET DATE DE SOUTENANCE

Au moment de l'annonce de soutenance, le lieu et la date de soutenance, servent à donner au public toutes les informations nécessaires pour assister à la soutenance. Étant donné que les soutenances de doctorat doivent être publiques. Il faut donc une information détaillée permettant au public d'y accéder, précisant ainsi l'horaire de début de la soutenance, la salle, l'adresse physique en présentiel ou le lien d'accès à la salle virtuelle lorsque la soutenance se tient en visioconférence ou les deux.

En revanche, **pour la couverture de la thèse** et le dépôt légal de la thèse, le lieu et la date de soutenance ont une fonction « légale » : le lieu définit de quelle juridiction relève le dépôt légal de la thèse et la date est utile, par exemple pour définir l'antériorité ou la fin d'une période de confidentialité. L'information doit donc être donnée sous une forme beaucoup plus synthétique que dans l'annonce de soutenance.

Sur la couverture de la thèse, la date doit être fournie au format « **JJ Mois AAA** » et le lieu de soutenance est simplement la ville, la commune ou la communauté d'agglomérations où s'est tenue la soutenance. Lorsque la soutenance a eu lieu dans les locaux de l'université Paris-Saclay, le lieu à indiquer est celui de la communauté d'agglomérations où se trouve le siège de l'université Paris-Saclay, à savoir « **Paris-Saclay** », que la thèse ait eu lieu en présentiel ou en visioconférence.

Exemple : Thèse soutenue à Paris-Saclay, le 10 Mars 2021.

9 - CIVILITÉ, FÉMINISATION DES TITRES ET FONCTIONS

Il est recommandé de ne pas indiquer les civilités (Madame / Monsieur) ni pour le docteur ou la docteure, ni pour les membres du Jury ou de l'équipe d'encadrement.

Toutefois, si cette recommandation n'était pas suivie, il faudrait alors assurer l'homogénéité. La civilité devrait alors être précisée pour **toutes les personnes** qui figurent sur la couverture (docteur.e, membres du Jury ou de l'équipe d'encadrement) en utilisant les **mêmes conventions** pour tous (Madame / Monsieur ou Mme / M.)

Il est recommandé de féminiser les titres des membres du Jury ou de l'équipe d'encadrement (Professeur / Professeure, Maître ou Maîtresse de conférences etc.) ainsi que les fonctions tenues dans le Jury (examineur / examinatrice ou Présidente / Présidente).

Pour « rapporteur », la forme féminine n'est pas recommandée faute de stabilisation. Si la personne concernée souhaitait la forme féminine, il faudrait alors lui demander de préciser la forme (« rapporteure » ou « rapporteuse »?) qu'elle préfère voir figurer sur la couverture.

10 - PRÉSENTATION DE LA DIRECTION DE LA THÈSE OU DE L'ÉQUIPE D'ENCADREMENT

Toutes les informations sur la direction de la thèse et l'équipe d'encadrement sont précisées sur la couverture de thèse (par exemple : directeur ou directrice de thèse, co-directeur ou codirectrice de thèse, co-encadrant ou co-encadrante, le cas échéant tuteur ou tutrice ou superviseur en entreprise).

Il faut également faire figurer lisiblement, à ce niveau, leur rôle vis-à-vis du doctorant ou de la doctorante et dans la préparation de la thèse.

Le directeur de thèse ou la directrice de thèse est cité en premier et est suivi, le cas échéant, des autres membres de l'équipe d'encadrement de la thèse, co-directeur ou co-directrice par ordre alphabétique puis des co-encadrantes et co-encadrantes par ordre alphabétique, puis tuteur ou tutrices par ordre alphabétique.

Le directeur ou la directrice de thèse et toute autre personne ayant participé à la direction scientifique des travaux et à l'encadrement du doctorant ou de la doctorante ne prend pas part à la décision de Jury de soutenance de doctorat. Ils et elles ne sont ni président, ni rapporteurs, ni examinateurs dans le Jury. Aucun membre de l'équipe d'encadrement ne fait partie des membres du Jury avec voix délibérative.

Puisqu'ils et elles n'apparaissent pas via le Jury, il est essentiel que leur rôle soit précisé clairement et lisiblement, sur la couverture de la thèse et sur www.theses.fr pour leur rôle dans l'équipe de direction de la thèse. Les autres personnes qui ont pu contribuer significativement à la thèse sans faire partie de l'équipe d'encadrement, ni du Jury, sont signalées dans la page des remerciements.

11 - COMPOSITION DU JURY

La soutenance de la thèse est une évaluation. Les travaux de recherche de doctorat devant être originaux à l'échelle internationale, le Jury est composé sur mesure pour chaque docteur.e et chaque thèse de doctorat. La composition du Jury est essentielle, le doctorat est délivré par l'université, sous condition du dépôt légal de la thèse, sur avis conforme du Jury. **Le Jury est garant de la qualité de la thèse.**

Pour chacun des membres du Jury, il faut préciser le **titre**, l'**affiliation** et la **fonction dans le jury** sur la page de couverture.

11.1 . A QUOI SERVENT CES INFORMATIONS ?

Ces informations doivent permettre, au premier regard, de vérifier la **conformité de la composition** du Jury :

- sa **légitimité** académique pour se prononcer sur l'obtention du plus haut diplôme universitaire, le doctorat (le jury comprend au moins la moitié de professeurs et assimilés et, sauf dérogation, les membres du Jury sont tous eux-mêmes docteurs).
- sa capacité à se prononcer en toute **indépendance** (au moins la moitié d'externes, à l'établissement de soutenance, à l'école doctorale, à l'équipe d'encadrement, au projet doctoral).

11.2 . LÉGITIMITÉ ACADÉMIQUE

Les **titres** des membres du Jury permettent de vérifier **qu'au moins la moitié des membres du Jury est professeur** des universités ou assimilé.

Les libellés exacts des titres français assimilés aux professeurs des universités (au moins la moitié du Jury) sont disponibles sur [legifrance](https://www.legifrance.gouv.fr/).

Le **président du Jury** est obligatoirement professeur des universités ou assimilé¹.

1. **Arrêté du 15 juin 1992** fixant la liste des corps de fonctionnaires assimilés aux professeurs des universités et aux maîtres de conférences pour la désignation des membres du Conseil national des universités : <https://www.legifrance.gouv.fr/affichTexte.do?cidTexte=LEGITEXT000019860291>

Arrêté du 10 février 2011 relatif à la grille d'équivalence des titres, travaux et fonctions des enseignants-chercheurs mentionnée aux articles 22 et 43 du décret n° 84-431 du 6 juin 1984 fixant les dispositions statutaires communes applicables aux enseignants-chercheurs et portant statut particulier du corps des professeurs des universités et du corps des maîtres de conférences : https://www.galaxie.enseignementsup-recherche.gouv.fr/ensup/pdf/EC_pays_etrangers/Tableau_comparaison_au_26_septembre_2012.pdf

Si l'un des rapporteurs n'était pas professeur des universités ou assimilé, il faudrait alors préciser, en plus, qu'il dispose bien de l'HDR (par exemple : Maître de conférences, HDR).

11.3 . INDÉPENDANCE

Les affiliations permettent de vérifier que le Jury est bien en **majorité externe** à l'établissement de soutenance, à l'école doctorale et à l'équipe d'encadrement. Pour cela, le nom ou l'acronyme du laboratoire ne suffit pas, en revanche, le nom de l'université ou de l'établissement délivrant le doctorat de rattachement du membre du jury suffit. Il n'est pas utile de préciser certains détails comme l'adresse postale complète ou le pays.

Exemple d'affiliation inadaptée car ambiguë : « IJCLab »

Lorsque le membre du jury est un chercheur d'un organisme national, fournir le nom de son organisme de rattachement ne suffit pas pour juger de son extériorité (CNRS par exemple). Dans ce cas-là ou dans d'autres cas où il y aurait une incertitude de cette nature, susceptible de susciter des interrogations sur le fait qu'au moins la moitié des membres du Jury est externe, il est alors demandé de préciser, en plus, l'université ou l'établissement où ce chercheur inscrit habituellement ses propres doctorants.

Exemple d'affiliation inadaptée car ambiguë : « CNRS »

Exemple d'affiliation adaptée : « CNRS, Université de Toulouse »

Lorsqu'il s'agit d'une entreprise ou d'une fondation ou d'une organisation qui n'est pas en lien direct avec un établissement d'enseignement supérieur pour l'inscription de doctorants, il faut alors le préciser.

Par exemple : « Saint Gobain recherche, entreprise »

Par exemple : « Moveo, Pôle de compétitivité »

11.4 . FONCTION DANS LE JURY ET ORDRE DE CITATION

La fonction dans le Jury de chaque membre du Jury doit également être précisée sur la page de couverture.

Les fonctions possibles dans un Jury sont : président(e), examinateur ou examinatrice, rapporteur et directeur ou directrice de thèse.

11.4.1 . Ordre de citation

Le président du Jury est le premier de la liste. Il est immédiatement suivi des deux rapporteurs dans l'ordre alphabétique, puis des autres examinateurs dans l'ordre alphabétique.

Un membre du Jury peut avoir deux fonctions dans le Jury (par ex. rapporteur & examinateur).

11.4.2 . Les rapporteurs

Les rapporteurs participent à l'évaluation de la thèse et figurent donc sur la couverture de thèse, qu'ils soient présents ou non le jour de la soutenance.

Si un rapporteur était absent le jour de la soutenance, il figurerait alors en tant que rapporteur seulement, sinon il figure à la fois en tant que rapporteur & examinateur. Lorsqu'un membre du Jury autre qu'un rapporteur, n'a pas pu participer au Jury de soutenance, physiquement ou bien en visioconférence, son nom ne figure pas sur la page de couverture de la thèse. Dans ce cas, il faut veiller à ce que la composition du Jury reste conforme, malgré l'absence du membre du Jury désigné. Cela peut demander de faire passer un membre interne en invité, par exemple.

12 - BIEN CITER SES SOURCES

La citation des sources fait partie intégrante du travail scientifique et participe de sa qualité et de son intégrité.

12.1 . S'INFORMER SUR LE PLAGIAT

Des mauvaises pratiques de citation peuvent conduire, même sans le vouloir, au plagiat. Le copier/coller, la paraphrase, la réutilisation d'images ou d'idées sans citer la source sont des situations de plagiat (n'hésitez pas à regarder cette courte vidéo sur les différentes formes de plagiat, volontaires ou non : <https://infotrack.unige.ch/comment-reconnaitre-les-cas-de-plagiat>)

Chaque discipline possède ses propres normes en termes de citation des sources. Renseignez-vous auprès de vos pairs pour connaître le style bibliographique et le style de citation à privilégier. Nous vous encourageons vivement à utiliser un logiciel de gestion bibliographique tel que Zotero. Vous pouvez retrouver des supports de formation à ce logiciel dans l'espace eCampus Doctorat, ouvert à tou-te-s sur auto-inscription : <https://ecampus.paris-saclay.fr/course/view.php?id=36678>

12.2 . LES IMAGES

Vous pouvez réutiliser dans votre thèse des images provenant d'articles ou de livres protégés par un copyright. Cela relève en effet de l'exception pédagogique, une des exceptions au droit d'auteur. Attention cependant, vous ne pouvez pas faire ce que vous voulez de cette image ! La loi vous autorise à inclure jusqu'à 20 images (en 720 dpi) sans demander d'autorisation à l'auteur. En revanche, une autorisation est nécessaire à partir de la 21^e image. Les sources des images doivent être mentionnées et aucune modification n'est autorisée.

Pour une présentation détaillée des différents cas d'utilisation d'images dans les thèses et les travaux universitaires, voir : <https://ethiquedroit.hypotheses.org/2947>

12.3 . ARTICLES JOINTS A LA THÈSE

Vous pouvez joindre vos articles à votre thèse. Toutefois, si votre thèse est diffusée en ligne (tout de suite après la soutenance, après un embargo ou après une période de confidentialité), il convient de s'assurer que vous respectez bien les politiques des éditeurs. En effet, tous n'autorisent pas la diffusion en accès libre de la version éditeur des articles. Utilisez [Sherpa Romeo](#) pour connaître la politique des éditeurs.

Selon la [Loi pour une république numérique](#), si votre recherche est financée à au moins 50% par des fonds publics français, vous avez le droit, en tant qu'auteur, de diffuser la version acceptée de l'article (mais sans la mise en pages de l'éditeur) au bout de 6 mois après publication pour les articles en sciences, techniques et médecine et 12 mois pour les articles en sciences humaines et sociales. Et ce quelque soit la politique éditoriale de l'éditeur.

Pour plus d'informations sur cette question, consultez la section « Déposer dans une archive ouverte » du [Passeport pour la science ouverte](#).

13 - DÉPOSER ET DIFFUSER SA THÈSE

La thèse fait l'objet d'un dépôt légal, en deux étapes, avant la remise du manuscrit aux rapporteurs et après la soutenance, qui protège le droit d'auteur du docteur.

Elle fait ensuite l'objet d'une diffusion sur le portail national des thèses www.theses.fr et le portail européen des thèses [DART-Europe](http://www.dart-europe.eu), sauf si la thèse présente un caractère confidentiel avéré.

13.1 . LES RESSOURCES A CONSULTER

- [Je publie, Quels sont mes droits ?](#)
- Le cadre réglementaire du [dépôt légal](#) sur Légifrance

13.2 . LES DÉMARCHES

Retrouvez le détail des démarches du dépôt de thèse dans cette [fiche explicative](#)

Si la thèse présente un caractère confidentiel avéré, le classement confidentiel de la thèse et, si nécessaire, une dérogation au caractère public de la soutenance (huis-clos) peuvent être [demandés au chef d'établissement](#).

14 - ANNEXE : LES LOGOS INSTITUTIONNELS

14.1 . LE LOGO DE L'UNIVERSITÉ PARIS-SACLAY



14.2 . LOGOS, NUMÉROS D'ACCRÉDITATION ET DÉNOMINATIONS DES ÉCOLES DOCTORALES

◆ n°127 : astronomie et astrophysique d'Île-de-France (AAIF)



◆ n°129 : sciences de l'environnement d'Île-de-France (SEIF)



◆ n°564 : physique en Île-de-France (PIF)



◆ n°566 : sciences du sport, de la motricité et du mouvement humain (SSMMH)



◆ n°567 : sciences du végétal : du gène à l'écosystème (SEVE)



ÉCOLE DOCTORALE

Sciences du végétal :
du gène à l'écosystème
(SEVE)

◆ n°568 : signalisations et réseaux intégratifs en biologie (Biosigne)



ÉCOLE DOCTORALE

Signalisations et réseaux
intégratifs en biologie
(BIOSIGNE)

◆ n°569 : innovation thérapeutique : du fondamental à l'appliqué (ITFA)



ÉCOLE DOCTORALE

Innovation thérapeutique
du fondamental à l'appliqué
(ITFA)

◆ n°570 : santé publique (EDSP)



ÉCOLE DOCTORALE

Santé Publique
(EDSP)

◆ n°571 : sciences chimiques : molécules, matériaux, instrumentation et bio-systèmes (2MIB)



ÉCOLE DOCTORALE

Sciences Chimiques: Molécules,
Matériaux, Instrumentation
et Biosystèmes (2MIB)

◆ n°572 : ondes et matière (EDOM)



ÉCOLE DOCTORALE

Ondes et matière
(EDOM)

◆ n°573 : interfaces : matériaux, systèmes, usages (INTERFACES)



ÉCOLE DOCTORALE

Interfaces:
matériaux, systèmes, usages

◆ n°574 : mathématiques Hadamard (EDMH)



ÉCOLE DOCTORALE

de mathématiques
Hadamard (EDMH)

◆ n°575 : electrical, optical, bio : physics and engineering (EOBE)



ÉCOLE DOCTORALE

Physique et ingénierie:
Electrons, Photons,
Sciences du vivant (EOBE)

◆ n°576 : particules hadrons énergie et noyau : instrumentation, imagerie, cosmos et simulation (PHENIICS)



ÉCOLE DOCTORALE

Particules, hadrons, énergie et noyau:
instrumentation, imagerie, cosmos
et simulation (PHENIICS)

◆ n°577 : structure et dynamique des systèmes vivants (SDSV)



ÉCOLE DOCTORALE

Structure et dynamique
des systèmes vivants
(SDSV)

◆ n°579 : sciences mécaniques et énergétiques, matériaux et géosciences (SMEMaG)



ÉCOLE DOCTORALE

Sciences mécaniques et
énergétiques, matériaux
et géosciences (SMEMAG)

◆ n°580 : sciences et technologies de l'information et de la communication (STIC)



ÉCOLE DOCTORALE

Sciences et technologies
de l'information et de
la communication (STIC)

◆ n°581 : agriculture, alimentation, biologie, environnement, santé (ABIES)



ÉCOLE DOCTORALE

Agriculture, alimentation,
biologie, environnement,
santé (ABIES)

◆ n°582 : cancérologie : biologie - médecine - santé (CBMS)



ÉCOLE DOCTORALE

Cancérologie : biologie -
médecine - santé (CBMS)

◆ n°629 : Sciences sociales et humanités (SSH)



ÉCOLE DOCTORALE

Sciences Sociales
et Humanités (SSH)

◆ n°630 : Droit, Économie, Management (DEM)



ÉCOLE DOCTORALE

Droit, Économie,
Management (DEM)

Bibliographie

- [1] James B. Tsui. *Digital techniques for wideband receivers*. T. 2. Sci-tech Publishing, 2004. url : https://books.google.com/books?hl=fr&lr=&id=GJluaAtwa8QC&oi=fnd&pg=PR17&dq=Digital+Techniques+for+Wideband+Receivers&ots=xmqj5zEhzG&sig=BeYqle_LSrToo-KsvwNBaZtY4DQ (visité le 09/12/2025).
- [2] Palghat P. Vaidyanathan et Piya Pal. "Sparse sensing with co-prime samplers and arrays". In : *IEEE Transactions on Signal Processing* 59.2 (2010). Publisher : IEEE, p. 573-586. url : <https://ieeexplore.ieee.org/abstract/document/5609222/> (visité le 09/12/2025).
- [3] Xiaowei Li, Hong Liang et Xiang-Gen Xia. "A robust Chinese remainder theorem with its applications in frequency estimation from undersampled waveforms". In : *IEEE Transactions on Signal Processing* 57.11 (2009). Publisher : IEEE, p. 4314-4322. url : <https://ieeexplore.ieee.org/abstract/document/5071209/> (visité le 09/12/2025).
- [4] H. K. Mardia. "New techniques for the deinterleaving of repetitive sequences". In : *IEE Proceedings F-Radar and Signal Processing*. T. 136. Issue : 4. IET, 1989, p. 149-154. url : <https://ieeexplore.ieee.org/iel1/2208/5666/00216611.pdf> (visité le 09/12/2025).
- [5] David Adamy. *EW 101 : A first course in electronic warfare*. T. 101. Artech house, 2001. url : <https://books.google.com/books?hl=fr&lr=&id=RZ0vDwAAQBAJ&oi=fnd&pg=PR7&dq=EW+101:+A+First+Course+in+Electronic+Warfare&ots=2VLAyIDJtH&sig=l1g-52gjyAkkBcarWFFTYiIlZF8> (visité le 09/12/2025).
- [6] Michael Grieves. "Digital twin : manufacturing excellence through virtual factory replication". In : *White paper* 1.2014 (2014), p. 1-7.
- [7] Elisa Negri, Luca Fumagalli et Marco Macchi. "A review of the roles of digital twin in CPS-based production systems". In : *Procedia manufacturing* 11 (2017). Publisher : Elsevier, p. 939-948. url : <https://www.sciencedirect.com/science/article/pii/S2351978917304067> (visité le 26/11/2025).
- [8] Alexey Dosovitskiy et al. "CARLA : An open urban driving simulator". In : *Conference on robot learning*. PMLR, 2017, p. 1-16. url : <https://proceedings.mlr.press/v78/dosovitskiy17a.html> (visité le 26/11/2025).

- [9] Fei Tao et al. "Digital twin in industry : State-of-the-art". In : *IEEE Transactions on industrial informatics* 15.4 (2018). Publisher : IEEE, p. 2405-2415. url : <https://ieeexplore.ieee.org/abstract/document/8477101/> (visité le 26/11/2025).
- [10] William R. Sherman et Alan B. Craig. *Understanding virtual reality : Interface, application, and design*. Morgan Kaufmann, 2018. url : <https://books.google.com/books?hl=fr&lr=&id=D-0cBAAQBAJ&oi=fnd&pg=PP1&dq=Understanding+Virtual+Reality+:+Interface,+Application,+and+Design&ots=QT0icdfR4U&sig=7NVN02ZhXIV7ehae-by0E-2w5u0> (visité le 26/11/2025).
- [11] Peter Fritzson. *Principles of object-oriented modeling and simulation with Modelica 3.3 : a cyber-physical approach*. John Wiley & Sons, 2015. url : https://books.google.com/books?hl=fr&lr=&id=wgIaBgAAQBAJ&oi=fnd&pg=PR13&dq=Principles+of+Object-Oriented+Modeling+and+Si-mulation+with+Modelica&ots=cZ60scKEkN&sig=SxTVWzN56d47byaUV6l_Qq0vZoE (visité le 26/11/2025).
- [12] Greg Brockman et al. *OpenAI Gym*. 5 juin 2016. doi : 10.48550/arXiv.1606.01540. arXiv : 1606.01540[cs]. url : <http://arxiv.org/abs/1606.01540> (visité le 26/11/2025).
- [13] Ben Mildenhall et al. "NeRF : representing scenes as neural radiance fields for view synthesis". In : *Communications of the ACM* 65.1 (jan. 2022), p. 99-106. issn : 0001-0782, 1557-7317. doi : 10.1145/3503250. url : <https://dl.acm.org/doi/10.1145/3503250> (visité le 26/11/2025).
- [14] Thomas Müller et al. "Instant neural graphics primitives with a multiresolution hash encoding". In : *ACM Transactions on Graphics* 41.4 (juill. 2022), p. 1-15. issn : 0730-0301, 1557-7368. doi : 10.1145/3528223.3530127. url : <https://dl.acm.org/doi/10.1145/3528223.3530127> (visité le 26/11/2025).
- [15] Bernhard Kerbl et al. "3D Gaussian splatting for real-time radiance field rendering." In : *ACM Trans. Graph.* 42.4 (2023), p. 139-1. url : [https://sgvr.kaist.ac.kr/~sungeui/ICG_F23/Students/\[CS482\]%203D%20Gaussian%20Splatting%20for%20Real-Time%20Radiance%20Field%20Rendering.pdf](https://sgvr.kaist.ac.kr/~sungeui/ICG_F23/Students/[CS482]%203D%20Gaussian%20Splatting%20for%20Real-Time%20Radiance%20Field%20Rendering.pdf) (visité le 26/11/2025).
- [16] Ben Poole et al. *DreamFusion : Text-to-3D using 2D Diffusion*. 29 sept. 2022. doi : 10.48550/arXiv.2209.14988. arXiv : 2209.14988[cs]. url : <http://arxiv.org/abs/2209.14988> (visité le 26/11/2025).

- [17] Zhengyi Wang et al. "Prolificdreamer : High-fidelity and diverse text-to-3d generation with variational score distillation". In : *Advances in neural information processing systems* 36 (2023), p. 8406-8441. url : https://proceedings.neurips.cc/paper_files/paper/2023/hash/1a87980b9853e84dfb295855b425c262-Abstract-Conference.html (visité le 26/11/2025).
- [18] Alvaro Sanchez-Gonzalez et al. "Learning to simulate complex physics with graph networks". In : *International conference on machine learning*. PMLR, 2020, p. 8459-8468. url : <https://proceedings.mlr.press/v119/sanchez-gonzalez20a> (visité le 26/11/2025).
- [19] Maziar Raissi, Paris Perdikaris et George E. Karniadakis. "Physics-informed neural networks : A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". In : *Journal of Computational physics* 378 (2019). Publisher : Elsevier, p. 686-707. url : <https://www.sciencedirect.com/science/article/pii/S0021999118307125> (visité le 26/11/2025).
- [20] Samuel Greydanus, Misko Dzamba et Jason Yosinski. "Hamiltonian neural networks". In : *Advances in neural information processing systems* 32 (2019). url : <https://proceedings.neurips.cc/paper/2019/hash/26cd8ecadce0d4efd6cc8a8725cbd1f8-Abstract.html> (visité le 26/11/2025).
- [21] Zongyi Li et al. *Fourier Neural Operator for Parametric Partial Differential Equations*. 17 mai 2021. doi : 10.48550/arXiv.2010.08895. arXiv : 2010.08895[cs]. url : <http://arxiv.org/abs/2010.08895> (visité le 26/11/2025).
- [22] David Silver et al. "Mastering the game of go without human knowledge". In : *nature* 550.7676 (2017). Publisher : Nature Publishing Group UK London, p. 354-359. url : https://idp.nature.com/authorize/casa?redirect_uri=https://www.nature.com/articles/nature24270&casa_token=uxVzaLwHPRIAAAAA:ABf8yG3mit_-tnl5ZCgrjrpH2A_BC18nsu5zAdIGdvnCQ2HUA0cQqPyyGJEWgDg4MSMrH4DvTYUcfmSITqw (visité le 26/11/2025).
- [23] Christopher R. DeMay et al. "Alphadogfight trials : Bringing autonomy to air combat". In : *Johns Hopkins APL Technical Digest* 36.2 (2022), p. 154-163. url : <https://secwww.jhuapl.edu/techdigest/Content/techdigest/pdf/V36-N02/36-02-DeMay.pdf> (visité le 26/11/2025).

- [24] Josh Tobin et al. "Domain randomization for transferring deep neural networks from simulation to the real world". In : *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017, p. 23-30. url : <https://ieeexplore.ieee.org/abstract/document/8202133/> (visité le 26/11/2025).
- [25] Rui Wang et al. *Paired Open-Ended Trailblazer (POET) : Endlessly Generating Increasingly Complex and Diverse Learning Environments and Their Solutions*. 21 fév. 2019. doi : 10.48550/arXiv.1901.01753. arXiv : 1901.01753[cs]. url : <http://arxiv.org/abs/1901.01753> (visité le 26/11/2025).
- [26] Diederik P. Kingma et Max Welling. "Auto-encoding variational bayes". In : *arXiv preprint arXiv :1312.6114* (2013). url : https://indico.math.cnrs.fr/event/11377/attachments/4589/6915/18012024_Kingma-and-Welling-2022%20Auto-Encoding%20Variational%20Bayes.pdf (visité le 26/11/2025).
- [27] Kihyuk Sohn, Honglak Lee et Xinchen Yan. "Learning structured output representation using deep conditional generative models". In : *Advances in neural information processing systems* 28 (2015). url : <https://proceedings.neurips.cc/paper/2015/hash/8d55a249e6baa5c06772297520da2051-Abstract.html> (visité le 26/11/2025).
- [28] Ali Razavi, Aaron Van den Oord et Oriol Vinyals. "Generating diverse high-fidelity images with vq-vae-2". In : *Advances in neural information processing systems* 32 (2019). url : <https://proceedings.neurips.cc/paper/2019/hash/5f8e2fa1718d1bbcadf1cd9c7a54fb8c-Abstract.html> (visité le 26/11/2025).
- [29] David Ha et Jürgen Schmidhuber. "World models". In : *arXiv preprint arXiv :1803.10122* 2.3 (2018). url : https://www.cl.cam.ac.uk/~ey204/teaching/ACS/R244_2024_2025/presentation/S6/WM_Edmund.pdf (visité le 26/11/2025).
- [30] Ian Goodfellow et al. "Generative adversarial networks". In : *Communications of the ACM* 63.11 (22 oct. 2020), p. 139-144. issn : 0001-0782, 1557-7317. doi : 10.1145/3422622. url : <https://dl.acm.org/doi/10.1145/3422622> (visité le 26/11/2025).
- [31] Shakir Mohamed et Balaji Lakshminarayanan. *Learning in Implicit Generative Models*. 27 fév. 2017. doi : 10.48550/arXiv.1610.03483. arXiv : 1610.03483[stat]. url : <http://arxiv.org/abs/1610.03483> (visité le 26/11/2025).

- [32] Phillip Isola et al. "Image-to-image translation with conditional adversarial networks". In : *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, p. 1125-1134. url : http://openaccess.thecvf.com/content_cvpr_2017/html/Isola_Image-To-Image_Translation_With_CVPR_2017_paper.html (visité le 26/11/2025).
- [33] You Xie et al. "tempoGAN : a temporally coherent, volumetric GAN for super-resolution fluid flow". In : *ACM Transactions on Graphics* 37.4 (31 août 2018), p. 1-15. issn : 0730-0301, 1557-7368. doi : [10.1145/3197517.3201304](https://doi.org/10.1145/3197517.3201304). url : <https://dl.acm.org/doi/10.1145/3197517.3201304> (visité le 26/11/2025).
- [34] Jascha Sohl-Dickstein et al. "Deep unsupervised learning using nonequilibrium thermodynamics". In : *International conference on machine learning*. pmlr, 2015, p. 2256-2265. url : <http://proceedings.mlr.press/v37/sohl-dickstein15.html> (visité le 26/11/2025).
- [35] Jonathan Ho, Ajay Jain et Pieter Abbeel. "Denoising diffusion probabilistic models". In : *Advances in neural information processing systems* 33 (2020), p. 6840-6851. url : <https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html> (visité le 26/11/2025).
- [36] Jiaming Song, Chenlin Meng et Stefano Ermon. *Denoising Diffusion Implicit Models*. 5 oct. 2022. doi : [10.48550/arXiv.2010.02502](https://arxiv.org/abs/2010.02502). arXiv : [2010.02502\[cs\]](https://arxiv.org/abs/2010.02502). url : <http://arxiv.org/abs/2010.02502> (visité le 26/11/2025).
- [37] Alex Graves. *Generating Sequences With Recurrent Neural Networks*. 5 juin 2014. doi : [10.48550/arXiv.1308.0850](https://arxiv.org/abs/1308.0850). arXiv : [1308.0850\[cs\]](https://arxiv.org/abs/1308.0850). url : <http://arxiv.org/abs/1308.0850> (visité le 27/11/2025).
- [38] Ashish Vaswani et al. "Attention is all you need". In : *Advances in neural information processing systems* 30 (2017). url : <https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html> (visité le 26/11/2025).
- [39] Alec Radford et al. "Improving language understanding by generative pre-training". In : (2018). Publisher : San Francisco, CA, USA. url : <https://www.mikecaptain.com/resources/pdf/GPT-1.pdf> (visité le 26/11/2025).
- [40] Scott Reed et al. *A Generalist Agent*. 11 nov. 2022. doi : [10.48550/arXiv.2205.06175](https://arxiv.org/abs/2205.06175). arXiv : [2205.06175\[cs\]](https://arxiv.org/abs/2205.06175). url : <http://arxiv.org/abs/2205.06175> (visité le 26/11/2025).

- [41] Alexey Dosovitskiy. "An image is worth 16x16 words : Transformers for image recognition at scale". In : *arXiv preprint arXiv:2010.11929* (2020). url : <https://files.ryanccopley.com/Papers/2010.11929v2.pdf> (visit  le 26/11/2025).
- [42] Claude E. Shannon. "A mathematical theory of communication". In : *The Bell system technical journal* 27.3 (1948). Publisher : Nokia Bell Labs, p. 379-423. url : <https://ieeexplore.ieee.org/abstract/document/6773024/> (visit  le 01/12/2025).
- [43] George EP Box et al. *Time series analysis : forecasting and control*. John Wiley & Sons, 2015. url : <https://books.google.com/books?hl=fr&lr=&id=rNt5CgAAQBAJ&oi=fnd&pg=PR7&dq=Time+Series+Analysis+:+Forecasting+and+Control&ots=DL73yTjVXD&sig=ww98YPoC8MLPySQkm3Vsc01CtKI> (visit  le 01/12/2025).
- [44] A ron Van Den Oord, Nal Kalchbrenner et Koray Kavukcuoglu. "Pixel recurrent neural networks". In : *International conference on machine learning*. PMLR, 2016, p. 1747-1756. url : <https://proceedings.mlr.press/v48/oord16.html> (visit  le 01/12/2025).
- [45] David G. Lowe. "Distinctive Image Features from Scale-Invariant Keypoints". In : *International Journal of Computer Vision* 60.2 (1 r nov. 2004), p. 91-110. issn : 1573-1405. doi : 10.1023/B:VISI.0000029664.99615.94. url : <https://doi.org/10.1023/B:VISI.0000029664.99615.94> (visit  le 26/11/2025).
- [46] Herbert Bay, Tinne Tuytelaars et Luc Van Gool. "SURF : Speeded Up Robust Features". In : *Computer Vision – ECCV 2006*. Sous la dir. d'Ale  Leonardis, Horst Bischof et Axel Pinz. Berlin, Heidelberg : Springer, 2006, p. 404-417. isbn : 978-3-540-33833-8. doi : 10.1007/11744023_32.
- [47] N. Dalal et B. Triggs. "Histograms of oriented gradients for human detection". In : *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). T. 1. ISSN : 1063-6919. Juin 2005, 886-893 vol. 1. doi : 10.1109/CVPR.2005.177. url : <https://ieeexplore.ieee.org/abstract/document/1467360> (visit  le 26/11/2025).
- [48] Yann LeCun et al. "Gradient-based learning applied to document recognition". In : *Proceedings of the IEEE* 86.11 (2002). Publisher : IEEE, p. 2278-2324. url : <https://ieeexplore.ieee.org/abstract/document/726791/> (visit  le 26/11/2025).

- [49] Alex Krizhevsky, Ilya Sutskever et Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks". In : *Advances in neural information processing systems* 25 (2012). url : <https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html> (visit  le 26/11/2025).
- [50] Christian Szegedy et al. "Going deeper with convolutions". In : *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, p. 1-9. url : https://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Szegedy_Going_Deeper_With_2015_CVPR_paper.html (visit  le 26/11/2025).
- [51] Karen Simonyan et Andrew Zisserman. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 10 avr. 2015. doi : 10.48550/arXiv.1409.1556. arXiv : 1409.1556[cs]. url : <http://arxiv.org/abs/1409.1556> (visit  le 26/11/2025).
- [52] Kaiming He et al. "Deep residual learning for image recognition". In : *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, p. 770-778. url : http://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html (visit  le 26/11/2025).
- [53] Olaf Ronneberger, Philipp Fischer et Thomas Brox. "U-Net : Convolutional Networks for Biomedical Image Segmentation". In : *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Sous la dir. de Nassir Navab et al. T. 9351. Series Title : Lecture Notes in Computer Science. Cham : Springer International Publishing, 2015, p. 234-241. isbn : 978-3-319-24573-7 978-3-319-24574-4. doi : 10.1007/978-3-319-24574-4_28. url : http://link.springer.com/10.1007/978-3-319-24574-4_28 (visit  le 26/11/2025).
- [54] Forrest Iandola et al. "Densenet : Implementing efficient convnet descriptor pyramids". In : *arXiv preprint arXiv:1404.1869* (2014). url : <https://arxiv.org/abs/1404.1869> (visit  le 26/11/2025).
- [55] Aaron Van Den Oord et al. "Wavenet : A generative model for raw audio". In : *arXiv preprint arXiv:1609.03499* 12 (2016), p. 1. url : https://www.academia.edu/download/61836013/WAVENET_-_A_GENERATIVE_MODEL_FOR_RAW_AUDIO_-_1609.0349920200120-19152-1e9641f.pdf (visit  le 26/11/2025).
- [56] Jonas Gehring et al. "Convolutional sequence to sequence learning". In : *International conference on machine learning*. PMLR, 2017, p. 1243-1252. url : <https://proceedings.mlr.press/v70/gehring17a> (visit  le 26/11/2025).

- [57] Nal Kalchbrenner et al. *Neural Machine Translation in Linear Time*. 15 mars 2017. doi : [10.48550/arXiv.1610.10099](https://doi.org/10.48550/arXiv.1610.10099). arXiv : [1610.10099\[cs\]](https://arxiv.org/abs/1610.10099). url : <http://arxiv.org/abs/1610.10099> (visité le 26/11/2025).
- [58] Shaojie Bai. "An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling". In : *arXiv preprint arXiv:1803.01271* (2018).
- [59] Yi Tay et al. *Are Pre-trained Convolutions Better than Pre-trained Transformers?* 30 jan. 2022. doi : [10.48550/arXiv.2105.03322](https://doi.org/10.48550/arXiv.2105.03322). arXiv : [2105.03322\[cs\]](https://arxiv.org/abs/2105.03322). url : <http://arxiv.org/abs/2105.03322> (visité le 26/11/2025).
- [60] Du Tran et al. "Learning spatiotemporal features with 3d convolutional networks". In : *Proceedings of the IEEE international conference on computer vision*. 2015, p. 4489-4497. url : http://openaccess.thecvf.com/content_iccv_2015/html/Tran_Learning_Spatiotemporal_Features_ICCV_2015_paper.html (visité le 26/11/2025).
- [61] Fausto Milletari, Nassir Navab et Seyed-Ahmad Ahmadi. "V-net : Fully convolutional neural networks for volumetric medical image segmentation". In : *2016 fourth international conference on 3D vision (3DV)*. Ieee, 2016, p. 565-571. url : <https://ieeexplore.ieee.org/abstract/document/7785132/> (visité le 26/11/2025).
- [62] T. N. Kipf. "Semi-supervised classification with graph convolutional networks". In : *arXiv preprint arXiv:1609.02907* (2016). url : <https://bibbase.org/service/mendeley/bfbbf840-4c42-3914-a463-19024f50b30c/file/25dbdd06-4704-a33f-23d9-c626b08adc1e/160902907.pdf> (visité le 26/11/2025).
- [63] Will Hamilton, Zhitao Ying et Jure Leskovec. "Inductive representation learning on large graphs". In : *Advances in neural information processing systems* 30 (2017). url : <https://proceedings.neurips.cc/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7e9bea9-Abstract.html> (visité le 26/11/2025).
- [64] Charles Ruizhongtai Qi et al. "Pointnet++ : Deep hierarchical feature learning on point sets in a metric space". In : *Advances in neural information processing systems* 30 (2017). url : <https://proceedings.neurips.cc/paper/2017/hash/d8bf84be3800d12f74d8b05e9b89836f-Abstract.html> (visité le 26/11/2025).
- [65] Jeffrey L. Elman. "Finding Structure in Time". In : *Cognitive Science* 14.2 (mars 1990), p. 179-211. issn : 0364-0213, 1551-6709. doi : [10.1207/s15516709cog1402_1](https://doi.org/10.1207/s15516709cog1402_1). url : https://onlinelibrary.wiley.com/doi/10.1207/s15516709cog1402_1 (visité le 27/11/2025).

- [66] Sepp Hochreiter et Jürgen Schmidhuber. "Long short-term memory". In : *Neural computation* 9.8 (1997). Publisher : MIT press, p. 1735-1780. url : <https://ieeexplore.ieee.org/abstract/document/6795963/> (visité le 27/11/2025).
- [67] Kyunghyun Cho et al. *On the Properties of Neural Machine Translation : Encoder-Decoder Approaches*. 7 oct. 2014. doi : [10.48550/arXiv.1409.1259](https://doi.org/10.48550/arXiv.1409.1259). arXiv : [1409.1259\[cs\]](https://arxiv.org/abs/1409.1259). url : <http://arxiv.org/abs/1409.1259> (visité le 27/11/2025).
- [68] Junyoung Chung et al. *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*. 11 déc. 2014. doi : [10.48550/arXiv.1412.3555](https://doi.org/10.48550/arXiv.1412.3555). arXiv : [1412.3555\[cs\]](https://arxiv.org/abs/1412.3555). url : <http://arxiv.org/abs/1412.3555> (visité le 27/11/2025).
- [69] Albert Gu et al. "Hippo : Recurrent memory with optimal polynomial projections". In : *Advances in neural information processing systems* 33 (2020), p. 1474-1487. url : https://proceedings.neurips.cc/paper_files/paper/2020/hash/102f0bb6efb3a6128a3c750dd16729be-Abstract.html (visité le 27/11/2025).
- [70] Albert Gu, Karan Goel et Christopher Ré. *Efficiently Modeling Long Sequences with Structured State Spaces*. 5 août 2022. doi : [10.48550/arXiv.2111.00396](https://doi.org/10.48550/arXiv.2111.00396). arXiv : [2111.00396\[cs\]](https://arxiv.org/abs/2111.00396). url : <http://arxiv.org/abs/2111.00396> (visité le 27/11/2025).
- [71] Albert Gu et Tri Dao. "Mamba : Linear-time sequence modeling with selective state spaces". In : *First conference on language modeling*. 2024. url : <https://openreview.net/forum?id=tEYskw1VY2> (visité le 27/11/2025).
- [72] Ilya Sutskever, Oriol Vinyals et Quoc V. Le. "Sequence to sequence learning with neural networks". In : *Advances in neural information processing systems* 27 (2014). url : <https://proceedings.neurips.cc/paper/5346-sequence-to-sequence-learning-with-neural-> (visité le 27/11/2025).
- [73] Yonghui Wu et al. *Google's Neural Machine Translation System : Bridging the Gap between Human and Machine Translation*. 8 oct. 2016. doi : [10.48550/arXiv.1609.08144](https://doi.org/10.48550/arXiv.1609.08144). arXiv : [1609.08144\[cs\]](https://arxiv.org/abs/1609.08144). url : <http://arxiv.org/abs/1609.08144> (visité le 27/11/2025).
- [74] Edward Choi et al. "Doctor ai : Predicting clinical events via recurrent neural networks". In : *Machine learning for healthcare conference*. PMLR, 2016, p. 301-318. url : <http://proceedings.mlr.press/v56/Choi16> (visité le 27/11/2025).

- [75] Pantelis R. Vlachas et al. "Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks". In : *Proceedings of the Royal Society A : Mathematical, Physical and Engineering Sciences* 474.2213 (mai 2018), p. 20170844. issn : 1364-5021, 1471-2946. doi : [10.1098/rspa.2017.0844](https://doi.org/10.1098/rspa.2017.0844). url : <https://royalsocietypublishing.org/doi/10.1098/rspa.2017.0844> (visit  le 27/11/2025).
- [76] David Salinas et al. "DeepAR : Probabilistic forecasting with autoregressive recurrent networks". In : *International journal of forecasting* 36.3 (2020). Publisher : Elsevier, p. 1181-1191. url : <https://www.sciencedirect.com/science/article/pii/S0169207019301888> (visit  le 27/11/2025).
- [77] Sageev Oore et al. "This time with feeling : learning expressive musical performance". In : *Neural Computing and Applications* 32.4 (f v. 2020), p. 955-967. issn : 0941-0643, 1433-3058. doi : [10.1007/s00521-018-3758-9](https://doi.org/10.1007/s00521-018-3758-9). url : <http://link.springer.com/10.1007/s00521-018-3758-9> (visit  le 27/11/2025).
- [78] Dzmitry Bahdanau, Kyunghyun Cho et Yoshua Bengio. "Neural machine translation by jointly learning to align and translate". In : *arXiv preprint arXiv :1409.0473* (2014). url : <https://peerj.com/articles/cs-2607/code.zip> (visit  le 30/11/2025).
- [79] Oriol Vinyals, Meire Fortunato et Navdeep Jaitly. "Pointer networks". In : *Advances in neural information processing systems* 28 (2015). url : https://proceedings.neurips.cc/paper_files/paper/2015/hash/29921001f2f04bd3baee84a12e98098f-Abstract.html (visit  le 30/11/2025).
- [80] Jacob Devlin et al. "Bert : Pre-training of deep bidirectional transformers for language understanding". In : *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics : human language technologies, volume 1 (long and short papers)*. 2019, p. 4171-4186. url : https://aclanthology.org/N19-1423/?utm_campaign=The+Batch&utm_source=hs_email&utm_medium=email&_hsenc=p2ANqtz-_m9bbH_7ECE1h3lZ3D61TYg52rKpifVNjL4 (visit  le 30/11/2025).
- [81] Tom Brown et al. "Language models are few-shot learners". In : *Advances in neural information processing systems* 33 (2020), p. 1877-1901. url : https://proceedings.neurips.cc/paper_files/paper/2020/hash/1457c0d6bfc4967418bfb8ac142f64a-Abstract.html?utm_source=transaction&utm_medium=email&utm_campaign=linkedin_newsletter (visit  le 30/11/2025).

- [82] Colin Raffel et al. "Exploring the limits of transfer learning with a unified text-to-text transformer". In : *Journal of machine learning research* 21.140 (2020), p. 1-67. url : <http://www.jmlr.org/papers/v21/20-074.html> (visité le 30/11/2025).
- [83] Qingsong Wen et al. *Transformers in Time Series : A Survey*. 11 mai 2023. doi : [10.48550/arXiv.2202.07125](https://doi.org/10.48550/arXiv.2202.07125). arXiv : [2202.07125\[cs\]](https://arxiv.org/abs/2202.07125). url : <http://arxiv.org/abs/2202.07125> (visité le 30/11/2025).
- [84] Haoyi Zhou et al. "Informer : Beyond efficient transformer for long sequence time-series forecasting". In : *Proceedings of the AAAI conference on artificial intelligence*. T. 35. Issue : 12. 2021, p. 11106-11115. url : <https://ojs.aaai.org/index.php/AAAI/article/view/17325> (visité le 30/11/2025).
- [85] Haixu Wu et al. "Autoformer : Decomposition transformers with auto-correlation for long-term series forecasting". In : *Advances in neural information processing systems* 34 (2021), p. 22419-22430. url : <https://proceedings.neurips.cc/paper/2021/hash/bcc0d400288793e8bdcd7c19a8ac0c2b-Abstract.html> (visité le 30/11/2025).
- [86] Ailing Zeng et al. "Are transformers effective for time series forecasting?" In : *Proceedings of the AAAI conference on artificial intelligence*. T. 37. Issue : 9. 2023, p. 11121-11128. url : <https://ojs.aaai.org/index.php/AAAI/article/view/26317> (visité le 30/11/2025).
- [87] Yuqi Nie et al. *A Time Series is Worth 64 Words : Long-term Forecasting with Transformers*. 5 mars 2023. doi : [10.48550/arXiv.2211.14730](https://doi.org/10.48550/arXiv.2211.14730). arXiv : [2211.14730\[cs\]](https://arxiv.org/abs/2211.14730). url : <http://arxiv.org/abs/2211.14730> (visité le 30/11/2025).
- [88] Chen Sun et al. "Revisiting unreasonable effectiveness of data in deep learning era". In : *Proceedings of the IEEE international conference on computer vision*. 2017, p. 843-852. url : http://openaccess.thecvf.com/content_iccv_2017/html/Sun_Revisiting_Unreasonable_Effectiveness_ICCV_2017_paper.html (visité le 01/12/2025).
- [89] Ze Liu et al. "Swin transformer : Hierarchical vision transformer using shifted windows". In : *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, p. 10012-10022. url : https://openaccess.thecvf.com/content/ICCV2021/html/Liu_Swin_Transformer_Hierarchical_Vision_Transformer_Using_Shifted_Windows_ICCV_2021_paper (visité le 30/11/2025).

- [90] John Jumper et al. "Highly accurate protein structure prediction with AlphaFold". In : *nature* 596.7873 (2021). Publisher : Nature Publishing Group UK London, p. 583-589. url : <https://www.nature.com/articles/s41586-021-03819-2> (visité le 30/11/2025).
- [91] Lili Chen et al. "Decision transformer : Reinforcement learning via sequence modeling". In : *Advances in neural information processing systems* 34 (2021), p. 15084-15097. url : <https://proceedings.neurips.cc/paper/2021/hash/7f489f642a0ddb10272b5c31057f0663-Abstract.html> (visité le 30/11/2025).