JOURNAL AFRICAIN DES SCIENCES
VOL. 3, **11** (2026) 131-135

https://jafrisci.org

ISSN (online) : 3107-0949 ; ISSN (print) : 3107-0930

OPEN ACCESS

ARTICLE DE RECHERCHE

Article Info.:
Reçu : le 07/09/2026
Accepté : le 03/10/2026
Publié : le 05/10/2026

CONTRIBUTION A LA MODELISATION MATHEMATIQUE ET A L'ANALYSE DYNAMIQUE DES SYSTEMES D'INTELLIGENCE ARTIFICIELLE EN INGENIERIE MECATRONIQUE ET AUTOMATIQUE

Simon Guwa Gua Band Kidiamboko *, Gad Yimbu Katay

https://doi.org/10.70237/jafrisci.2026.v3.i11.11

Résumé

Cet article présente une modélisation mathématique et une analyse dynamique adaptées aux systems d'intelligence artificielle (IA) intégrés en ingénierie mécatronique et automatique. À travers l'articulation entre le Système Informatique (SI) et le Système Machine/Mécanisme (MM), l'étude formalise les interactions entre algorithmes décisionnels et processus physiques. En mobilisant les fonctions de transfert, la transformée de Laplace en temps continu et la transformée en Z en temps discret, le modèle permet d'évaluer la stabilité, la rapidité et la réactivité des algorithmes d'apprentissage. Une application sur la commande adaptative d'un moteur électrique illustre la synergie entre la théorie du contrôle et l'apprentissage automatique pour garantir des performances optimales face aux perturbations.

Mots-clés: Intelligence artificielle, Modélisation mathématique, Système dynamique, Fonction de transfert, Commande adaptative, Ingénierie mécatronique

Abstract

This paper presents a mathematical modeling and dynamic analysis framework tailored for artificial intelligence (AI) systems integrated into mechatronic and automatic control engineering. By linking the information System (IS) with the Machine/Mechanism System (MM), this study formalizes the interactions between decision-making algorithms and physical processes. Utilizing transfer functions, the continuous-time Laplace transform, and the discrete-time Z-transform, the proposed model evaluates the stability, speed, and responsiveness of learning algorithms. A practical case study on adaptive electric motor control demonstrates the synergy between control theory and machine learning to ensure robust performance under perturbations.

Keywords: Artificial intelligence, Mathematical modeling, Dynamic system, Transfer function, Adaptive control, Mechatronic Engineering

1. Introduction

1.1. Contexte et enjeux

Au XXIe siècle, l'intelligence artificielle(IA)s'est progressivement imposée comme un moteur essentiel d'innovation, touchant un large éventail de domaines : de la médecine aux transports, en passant par l'industrie.

Elle transforme en profondeur les pratiques d'ingénie rie en apportant des capacités nouvelles d'analyse, d'adaptation et d'automatisation. Dans le cadre précis de l'ingénierie mécatronique, du génie électrique et de l'automatique, discipline fondée sur la conception et la mise en œuvre de systèmes physiques instrumentés, cette évolution confronte les chercheurs à de nouveaux défis.

Comment assurer la robustesse, la stabilité et la fiabilité d'un système dont le comportement est issu d'un apprentissage automatique ? Comment relier un algorithme abstrait à sa mise en œuvre concrète sur une machine physique ? Et comment mesurer les performances d'un système hybride, à la fois logiciel et matériel, évoluant dans un environnement soumis à des perturbations réelles ? Ces questions prennent d'autant plus d'importance que l'IA est désormais intégrée à des systèmes critiques (robots industriels, véhicules autonomes, réseaux électriques intelligents), où la moindre défaillance peut avoir des conséquences graves : sécurité des usagers, bon fonctionnement d'infrastructures, respect des contraintes temporelles strictes. Cela impose le développement d'outils ma thématiques et conceptuels solides, capables d'analyser et de maîtriser le comportement dynamique de ces systèmes intelligents. C'est dans cette optique que s'inscrit l'étude mathématique des systèmes d'IA appliqués à l'ingénierie mécatronique : elle vise à proposer des modèles capables de

Correspondant : *s.kidiamboko@gmail.com* ***(S. G. G. B. Kidiamboko)***

Copyright : © The Author(s) Published under a Creative Commons Attribution 4.0 International Licence (CC BY 4.0)

Section Electronique, Institut Supérieur des Techniques Appliquées (ISTA), Kinshasa, République Démocratique du Congo

représenter leur dynamique, d'évaluer leur stabilité, leur précision ou encore leur capacité à s'adapter. Ces modèles deviennent alors des outils d'aide à la conception, au pilotage et à l'amélioration de systèmes embarqués.

1.2. Objectif de l'étude

Cette étude a pour ambition de proposer une modélisation mathématique claire et directement exploitable d'un système d'intelligence artificielle appliqué au domaine de l'ingénierie mécatronique et de la commande automatique. L'objectif est double : mieux comprendre le fonctionnement interne de ces systèmes, tout en fournissant aux ingénieurs des outils concrets pour leur conception et leur évaluation. Plus précisément, il s'agit de : - Définir une architecture globale articulant le Système Informatique (SI), qui exécute l'algorithme d'IA, et le Système Machine/Mécanisme (MM), qui traduit les décisions en actions physiques ;

- Représenter cette architecture à travers une modélisation dynamique fondée sur des équations différentielles et des fonctions de transfert, en tenant compte à la fois du temps continu (via la transformée de Laplace) et du temps discret (via la transformée en Z) ;
- Identifier et adapter des outils d'analyse pertinents tels que l'étude de la réponse fréquentielle, les critères de stabilité ou encore les indicateurs de performance aux caractéristiques propres des algorithmes d'apprentissage ;
- Illustrer cette démarche par une étude de cas simplifiée portant sur la régulation de vitesse d'un moteur électrique, accompagnée d'une simulation numérique

1.3. Méthodologie de modélisation

La démarche suivie dans cette étude repose sur une approche combinant rigueur mathématique et validation par la simulation numérique. Elle se décline en plusieurs étapes structurées :

- Structuration architecturale : définition des blocs fonctionnels principaux, le Système Informatique (SI) et le Système Machine/Mécanisme (MM), ainsi que leurs interactions en boucle ouverte ou fermée ;
- Modélisation dynamique : formulation de la dynamique globale par des équations différentielles transformées en fonctions de transfert en temps continu (variable $_p$) et en temps discret (variable $_z$) ;
- Analyse fréquentielle et temporelle : étude de la réponse dynamique pour évaluer la stabilité, la réactivité, l'amortissement et l'impact de la numérisation ;
- Simulation et étude de cas : mise en place d'un exemple numérique concret pour observer les courbes de réponse et valider le modèle ;
- Interprétation en ingénierie : traduction des résultats mathématiques en recommandations pratiques de conception et de réglage adaptatif.

2. Fondements théoriques

2.1 Définitions et typologies de l'intelligence artificielle

L'intelligence artificielle (IA) désigne l'ensemble des concepts, méthodes et technologies visant à permettre à des systèmes informatiques d'accomplir des tâches qui, en temps normal, nécessitent des capacités humaines : percevoir, comprendre le langage, planifier, apprendre ou encore prendre des decisions [1].

Comme le soulignent Goodfellow et al. [2], l'essence de l'IA réside dans sa capacity à apprendre à partir des données et à s'adapter à de nouvelles situations. Contrairement aux systèmes traditionnels qui suivent des instructions rigides, un système d'IA ajuste son comportement en fonction de l'expérience acquise.

Historiquement, deux grandes approches ont marqué le développement de l'IA :

- L'IA symbolique (ou classique), fondée sur des règles logiques explicites, servant de base aux systèmes experts [3].
- L'IA connexionniste, inspirée du fonctionnement cérébral, qui utilise des réseaux de neurones artificiels pour apprendre par l'expérience [4].

Aujourd'hui, on distingue deux types principaux d'IA :

1. L'IA faible (ou spécialisée) : conçue pour remplir une tâche précise (reconnaissance vocale, traduction, diagnostic) ;
2. L'IA forte (ou générale) : concept théorique visant à reproduire l'intelligence humaine globale.

De plus, des architectures hybrides dites neuro-symboliques émergent pour associer la logique formelle à l'apprentissage statistique [5].

2.2 Notions de système dynamique et de fonction de transfert

Dans le contexte de l'ingénierie, les systèmes intégrant de l'IA (robots autonomes, véhicules intelligents, régulateurs adaptatifs) se comportent comme des systèmes dynamiques dont l'état évolue dans le temps [6]. La fonction de transfert permet d'établir le rapport entre le signal d'entrée $x(t)$ et la sortie $y(t)$:

$$H(p) = \frac{Y(p)}{X(p)} = \frac{1}{ap^2 + bp + c} \quad (1)$$

Les coefficients $_a$, $_b$ et $_c$ représentent des paramètres dynamiques (inertie, frottement, raideur). L'étude s'appuie sur la transformée de Laplace pour le temps continu et la transformée en Z pour le temps discret des calculateurs numériques [7]. Le couplage global s'exprime par le produit des fonctions de transfert du Système Informatique $_{F(p)}$ et du Système Machine/Mécanisme $_{G(p)}$:

$$H(p) = F(p) \times G(p) \quad (2)$$

3. Modélisation mathématique d'un système d'IA

3.1. Structure générale du système (SI + MM)

Un système d'IA appliqué à l'ingénierie s'articule autour de deux composantes interconnectées :

1. Le Système Informatique (SI) : cœur logiciel exécutant les algorithmes de perception, d'apprentissage et de décision ;
2. Le Système Machine et Mécanisme (MM) : partie mécatronique physique (capteurs, actionneurs, moteurs) exécutant les commandes.

Cette boucle fermée permet une rétroaction continue nécessaire à l'apprentissage et à l'adaptation [8].

3.2. Équations différentielles et fonctions de transfert

3.2.1. Modèle du système global

Soit un système dynamique linéaire et invariant dans le temps (LTI) régi par l'équation différentielle :

$$a\frac{d^2y(t)}{dt^2} + b\frac{dy(t)}{dt} + cy(t) = x(t) \qquad (3)$$

En appliquant la transformée de Laplace sous conditions initiales nulles, nous obtenons :

$$(ap^2 + bp + c)Y(p) = X(p) \qquad (4)$$

D'où la fonction de transfert canonique :

$$H(p) = \frac{Y(p)}{X(p)} = \frac{1}{ap^2 + bp + c} \qquad (5)$$

Les paramètres agissent comme suit :

- a : inertie du système (vitesse de réponse) ;
- b : amortissement (dissipation des oscillations) ;
- c : raideur ou réactivité.

3.2.2 Couplage SI-MM

La fonction de transfert globale découle de l'association en série du logiciel $F(p)$ et du mécanisme $G(p)$:

$$Y(p) = F(p) \cdot G(p) \cdot X(p) \qquad (6)$$

Ce découplage facilite l'analyse séparée de l'algorithme d'apprentissage et de la dynamique physique [9].

3.3. Exemple numérique et simulation simplifiée

Considérons l'équation numérique suivante ($a = 1, b = 3, c = 2$) :

$$\frac{d^2y(t)}{dt^2} + 3\frac{dy(t)}{dt} + 2y(t) = x(t) \qquad (7)$$

La fonction de transfert associée s'écrit :

$$H(p) = \frac{1}{p^2 + 3p + 2} = \frac{1}{(p+1)(p+2)} \qquad (8)$$

Pour un signal d'impulsion, la réponse temporelle est :

$$y(t) = e^{-t} - e^{-2t} \qquad (9)$$

Le terme rapide e^{-2t} traduit la réaction algorithmique immédiate, tandis que le terme lent e^{-t} représente la phase d'adaptation progressive de l'IA.

3.4. Vers une modélisation discrète : transformée en Z

Pour les calculateurs embarqués à échantillonnage régulier, l'équation aux différences s'exprime par : $y[n] = a_1y[n-1] + a_2y[n-2] + b_0x[n]$ (10) Sa fonction de transfert en Z est donnée par :

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b_0}{1 - a_1z^{-1} - a_2z^{-2}} \qquad (11)$$

La stabilité numérique exige que tous les pôles de $H(z)$ soient situés à l'intérieur du cercle unité du plan complexe ($|z| < 1$).

4. Analyse et interpretation

4.1. Analyse temporelle : comportement dynamique et apprentissage

L'analyse temporelle sépare la réponse transitoire (adaptation initiale aux données) de la réponse permanente (stabilité de l'état appris). Elle évalue la rapidité de convergence et l'absence de dépassements destructeurs.

4.2. Analyse fréquentielle : stabilité et performance

4.2.1. Stabilité du système

Le critère BIBO (Bounded Input, Bounded Output) requiert que tous les pôles de $H(p)$ aient une partie réelle strictement négative[9], ou que les pôles de $H(z)$ restent strictly dans le cercle unité [6].

4.2.2. Performance et bande passante

Une bande passante élevée favorise une adaptation rapide mais accroît la sensibilité au bruit. Une bande passante étroite améliore la robustesse au détriment de la réactivité [4].

4.3. Signification technique pour l'IA

Le Tableau 1 synthétise la correspondance entre les grandeurs dynamiques et les propriétés de l'IA.

Tableau 1 – Correspondance entre propriétés dynamiques des systèmes et interprétation en IA

Propriété du système	Interprétation en IA	Objectif d'ingénierie
Stabilité (Pôles stables)	Convergence de l'algorithme sans divergence numérique	Garantir la sécurité et la fiabilité
Temps de réponse	Vitesse d'adaptation à de nouvelles données	Optimiser la réactivité globale
Amortissement	Contrôle du surapprentissage et des oscillations	Assurer la précision des décisions
Bande passante	Capacité à traiter la variabilité des données	Équilibrer stabilité et flexibilité

4.4. Perspectives analytiques

L'analyse des architectures profondes nécessite d'étendre ces concepts vers des systèmes multivariables, non linéaires (fonctions d'activation) et stochastiques [10].

5. Applications en ingénierie

5.1. L'IA comme outil d'optimisation et de contrôle intelligent

L'IA apporte une flexibilité inédite en commande adaptative, maintenance prédictive [11], gestion énergétique [12] et optimisation topologique [13].

5.2. Étude de cas pratique : système de contrôle intelligent d'un moteur

5.2.1. Description du système

Il s'agit de réguler la vitesse d'un moteur électrique en ajustant la commande via un contrôleur intelligent analysant l'écart $e(t) = x(t) - y(t)$.

5.2.2. Modélisation mathématique

La dynamique du moteur s'écrit :

$$G(p) = \frac{1}{J_p + b} \tag{12}$$

Le contrôleur PID adaptatif auto-ajusté par IA possède la fonction de transfert :

$$F(p) = K_p + \frac{K_i}{p} + K_{dp} \tag{13}$$

La fonction en boucle fermée devient :

$$H(p) = \frac{F(p)G(p)}{1 + F(p)G(p)} \tag{14}$$

Les gains K_p,K_i,K_d sont optimisés pour minimiser la fonction de coût :

$$J = \int_0^\infty [x(t) - y(t)]^2 dt \tag{15}$$

5.2.3. Interprétation

L'algorithme ajuste les gains en temps réel : il augmente K_p pour accélérer la convergence et ajuste K_d pour éliminer l'instabilité face aux variations de charge [14].

5.3. Discussion : apport de la modélisation mathématique dans l'IA appliquée

La modélisation offre une validation prédictive avant déploiement, optimise l'interopérabilité SI-MM et réduit le coût des essais physiques [15].

5.4. Perspectives d'application

Les domaines d'application privilégiés incluent la robotique autonome, les réseaux électriques intelligents, l'aéronautique [16] et l'industrie 4.0.

6. Conclusion générale et recommandations

6.1. Conclusion générale

Cette étude démontre l'efficacité du formalisme mathématique (Laplace, Z, fonctions de transfert) pour modéliser, analyser et sécuriser les systèmes d'intelligence artificielle appliqués à l'ingénierie. L'association de la théorie du contrôle et de l'apprentissage automatique garantit l'émergence de systèmes autonomes fiables et performants.

6.2. Recommandations

1. Intégrer systématiquement l'analyse fréquentielle lors du développement d'algorithmes d'IA embarqués ;
2. Privatiser les architectures hybrides combinant contrôle automatique (PID) et apprentissage par renforcement ;
3. Utiliser des outils de co-simulation (MATLAB/Simulink, Python) avant validation matérielle ;
4. Promouvoir une formation interdisciplinaire (automatique, mathématiques, IA) ;
5. Approfondir la recherche sur les dynamiques non linéaires et stochastiques.

6.3. Perspective finale

La formalisation mathématique transforme l'intelligence artificielle en un outil d'ingénierie mesurable, vérifiable et intrinsèquement sécurisé.

Références bibliographiques

[1] K. J. Åström et R. M. Murray, *Feedback Systems:AnIntroductionforScientistsand Engineers* (Princeton : Princeton University Press, 2010).
[2] I. Goodfellow, Y. Bengio, et A. Courville, *Deep Learning* (Cambridge : MIT Press, 2016).
[3] Newell et H. A. Simon, "Computer science as empirical inquiry : Symbols and search," *CommunicationsoftheACM*, 19, no. 3 (1976) : 113-126.
[4] Y. LeCun, Y. Bengio, et G. Hinton, "Deep learning," *Nature*, 521, no. 7553 (2015) : 436-444.
[5] T. R. Besold et al., "Neural-symbolic learning and reasoning : A survey and usage guide," *arXiv preprint arXiv:1711.03902*, 2017.
[6] K. J. Åström et R. M. Murray, *Feedback Systems:AnIntroductionforScientistsand Engineers* (Princeton : Princeton University Press, 2010).
[7] K. Ogata, *Discrete-Time Control Systems*, 2nd ed. (Upper Saddle River : Prentice Hall, 1995).
[8] R. S. Sutton et A. G. Barto, *Reinforcement Learning:AnIntroduction*, 2nd ed. (Cambridge : MIT Press, 2018).
[9] H. K. Khalil, *NonlinearControl* (New York : Pearson Education, 2015).
[10] E. D. Sontag, *MathematicalControlTheory: Deterministic Finite Dimensional Systems*, 2nd ed. (New York : Springer, 2013).
[11] J. Wang et al., "Deep learning for smart manufacturing : Methods and applications," *JournalofManufacturingSystems*, 48 (2018) : 144-156.
[12] Y. Zhao, Y. Ding, et Y. Zhang, "Artificial intelligence-based optimization for energy management systems," *Energy Reports*, 6 (2020) : 2653-2664.
[13] M. P. Bendsøe et O. Sigmund, *Topology Optimization : Theory, Methods, and Applications* (Berlin : Springer, 2013).
[14] T. Nguyen et S. Nahavandi, "Model-based reinforcement learning for adaptive control of dynamic systems," *IEEE Transactions on Cybernetics*, 50, no. 5 (2020) : 19781989.
[15] K. J. Åström et B. Wittenmark, *Adaptive Control*, 2nd ed. (Dover : Courier Corporation, 2013).
[16] S. Yin, H. Gao, et H. R. Karimi, "Data-driven monitoring and control : From process to smart manufacturing," *AnnualReviews in Control*, 51 (2021) : 317-331.