L'Apprentissage Machine au Sein des Jumeaux Numériques pour l'Estimation et la Détection des Menaces

Machine Learning in Digital Twins for Threat Estimation and Detection

Hugo Bourreau[1,2,3,5], Marc-Oliver Pahl[1,2,3], Fabien Dagnat[2,4] and Fehmi Jaafar[5]

[1]Chaire Cyber CNI, 2 Rue de la Châtaigneraie, 35510 Cesson-Sévigné
[2]IMT Atlantique, 2 Rue de la Châtaigneraie, 35510 Cesson-Sévigné, France
[3]Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA), 263 Av. Général Leclerc, 35000 Rennes
[4]Lab-STICC, UMR 6285, Brest, France
[5]Université du Québec à Chicoutimi (UQAC), 555 Bd de l'Université, Chicoutimi, QC G7H 2B1, Canada

Abstract

The convergence of Digital Twin (DT) technology and Machine Learning (ML) presents a promising combination for enhancing cybersecurity by proactively predicting threats and attacks. This paper proposes a comprehensive taxonomy of Digital Twins for cybersecurity, categorizing their roles in threat prediction and attack detection. By analyzing methodologies, feature selection strategies, and AI tools employed across implementations, it highlights their current capabilities and limitations.

Keywords

Cybersécurité, Jumeaux Numériques, Apprentissage Machine, Intelligence artificielle

Résumé

La convergence des jumeaux numériques (JN) et de l'apprentissage machine (ML) constitue une combinaison prometteuse pour améliorer la cybersécurité en prédisant proactivement les menaces et les attaques. Cet article présente une taxonomie de l'utilisation du concept des jumeaux numériques pour la cybersécurité, en catégorisant leurs rôles dans la prédiction des menaces et la détection des attaques. En analysant les méthodologies, les stratégies de sélection des caractéristiques des systèmes et les outils d'intelligence artificielle utilisés dans diverses implémentations, leurs capacités et limites actuelles sont mises en évidence.

1. Introduction

Avec l'augmentation des cybermenaces ciblant les infrastructures critiques, la cybersécurité des systèmes cyber-physiques (CPS) constitue une cible prioritaire en raison de leur rôle essentiel dans les domaines de l'énergie, du transport ou encore de l'industrie. Les Jumeaux Numériques (JN) apparaissent comme un levier prometteur pour renforcer les capacités de détection et d'anticipation des attaques. Leur couplage avec l'apprentissage machine (ML) ouvre de nouvelles perspectives pour modéliser, simuler, prédire et évaluer des scénarios d'attaque. Une revue systématique de 149 études a récemment synthétisé le rôle du ML au sein des JN, mettant en évidence une prédominance d'approches centrées sur le composant IA sans connexion robuste entre le système réel et son modèle virtuel [1]. Cette observation met en évidence la nécessité de concevoir des JN intégrant explicitement une interaction avec des données physiques. Pourtant, cette convergence soulève encore de nombreuses interrogations en matière de conception, d'adaptabilité et d'évaluation à long terme. C'est dans ce contexte que se positionne notre étude.

Cet article examine l'intégration de l'IA, et plus particulièrement le ML, dans les JN appliqués aux CPS pour renforcer la cybersécurité. Il commence par synthétiser les approches d'intégration du ML

C&ESAR'25: Computer & Electronics Security Application Rendezvous, Nov. 19-20, 2025, Rennes, France
hugo.bourreau@imt-atlantique.fr (H. Bourreau); marc-oliver.pahl@imt-atlantique.fr (M. Pahl); fabien.dagnat@imt-atlantique.fr (F. Dagnat); fjaafar@uqac.ca (F. Jaafar)
0009-0000-9877-1134 (H. Bourreau); 0000-0001-5241-3809 (M. Pahl); 0000-0002-2419-7587 (F. Dagnat); 0000-0002-4101-2281 (F. Jaafar)

CEUR Workshop Proceedings ISSN 1613-0073
published 2025-12-22

aux JN dans un contexte de cybersécurité, avec un accent particulier sur la détection d'anomalies et l'estimation des menaces. L'article présente d'abord les types de modèles d'apprentissage les plus utilisés, ainsi que les méthodes permettant leur intégration efficace dans les JN. Il décrit ensuite les mécanismes basés sur des modèles dynamiques et un apprentissage continu, permettant d'estimer en temps réel et à long terme les conséquences de cyberattaques sur les CPS.

En réponse à ces objectifs, les contributions de cet article sont les suivantes :

- Réalisation d'une synthèse des travaux de recherche existants sur l'intégration de l'apprentissage machine aux jumeaux numériques pour la cybersécurité.
- Analyse et classification des approches existantes combinant jumeaux numériques et apprentissage automatique en cybersécurité, et mise en évidence des défis associés.
- Analyse comparative des modèles d'intelligence artificielle couramment utilisés pour la détection d'anomalies.
- Identification et discussion des principaux défis liés à l'utilisation de l'apprentissage machine dans les jumeaux numériques.

Cet article adopte une démarche d'analyse structurée, reposant sur une revue ciblée des cadres d'implémentation récents et des approches expérimentales. La sélection des travaux s'est appuyée sur les bases de données scientifiques IEEE Xplore et ACM Digital Library, en se concentrant sur les publications parues entre 2020 et 2025, traitant de la convergence entre jumeaux numériques (JN), apprentissage machine (ML) et systèmes cyber-physiques (CPS) dans un contexte de cybersécurité.

Les contributions identifiées ont ensuite été analysées selon un ensemble de critères définis dans la taxonomie proposée en Section 2.4, tels que la nature du couplage JN-ML, l'horizon temporel d'analyse, le mode d'apprentissage, ou encore le rôle joué dans la chaîne de sécurité. Cette approche vise à fournir une grille de lecture cohérente permettant de faciliter leur comparaison.

La Section 2 présente le domaine étudié, en introduisant les concepts clés des JN et du ML, leur rôle en cybersécurité, ainsi qu'une taxonomie et une architecture de référence. La Section 3 est consacrée aux travaux existants, en détaillant les techniques mobilisées, les implémentations recensées et leur comparaison selon la taxonomie proposée. La Section 4 analyse les verrous actuels de l'intégration JN–ML et discute des approches émergentes visant à les dépasser. La Section 5 discute des principaux défis liés à l'intégration du ML au sein des JN pour l'estimation d'attaques. Enfin, la Section 6 conclut l'article et présente les perspectives de recherche futures.

2. Domaine

2.1. Les Jumeaux Numériques

Les jumeaux numériques (JN), apparus en 2002 et introduits par Grieves et al. [2], sont aujourd'hui devenus une composante essentielle de l'industrie 4.0. Ils visent à reproduire les caractéristiques essentielles d'un système de référence afin de permettre une surveillance et une analyse en temps réel. Un JN repose sur la combinaison de plusieurs technologies, incluant les capteurs et dispositifs IoT pour la collecte de données en temps réel, la modélisation physique et la simulation pour représenter le comportement du système, les infrastructures Cloud et Edge Computing pour assurer le traitement et la synchronisation, ainsi que des outils d'analyse avancée tels que les environnements Big Data et l'apprentissage machine. L'association de ces briques permet d'améliorer les capacités de supervision, d'optimisation et de prédiction du système modélisé. Le JN reflète en continu l'évolution du système, ce qui le rend utile pour la détection d'anomalies.

Toutefois, les JN traditionnels se limitent souvent à une modélisation déterministe, ce qui rend difficile la détection d'attaques jusque-là inconnues. L'intégration de ML permet d'exploiter les données quantitatives collectées par le JN pour identifier de nouveaux schémas contrairement aux approches purement basées sur des règles.

2.2. L'apprentissage machine

L'intelligence artificielle (IA) et plus particulièrement l'apprentissage machine (ML) sont de plus en plus intégrés dans les jumeaux numériques afin d'améliorer la puissance analytique et l'automatisation. Des techniques telles que l'apprentissage supervisé, l'apprentissage profond et les méthodes d'ensemble sont employées pour reconnaître des modèles, modéliser des séquences temporelles ou optimiser la prise de décision dans des environnements dynamiques. Les modèles sont entraînés par des jeux de données et leur qualité découle directement de celle de ces données d'entraînement. Un enjeu majeur est donc celui de la conception ou du choix du jeu de données d'entraînement.

Des recherches récentes ont démontré que la combinaison de ML et de JN permet non seulement la détection d'anomalies et d'intrusions [3], mais aussi la simulation de cyberattaques et l'évaluation de stratégies de réponse par le biais d'environnements virtuels [4] [5]. En intégrant du ML, le JN peut réagir à des anomalies connues et anticiper des comportements inédits.

2.3. Implication pour la cybersécurité

La convergence des JN et du ML transforme les pratiques de cybersécurité dans de nombreux domaines car elle permet de passer d'une logique réactive de détection à une approche proactive de prédiction et d'anticipation des menaces. En s'appuyant sur des modèles virtuels continuellement mis à jour et sur des analyses avancées basées sur des données, les organisations peuvent détecter, analyser et, dans certains cas, estimer les cybermenaces en temps réel avant même qu'elles n'impactent le système. Les JN dotés de ML peuvent simuler des scénarios d'attaque, identifier les comportements anormaux des systèmes et tester les mécanismes de défense dans un environnement virtuel isolé [4]. Ce qui implique un risque minimal pour les opérations réelles puisque les expérimentations et validations sont effectuées sur le modèle numérique plutôt que directement sur les infrastructures critiques. Cette intégration a conduit au développement de nouveaux cadres de sécurité pour l'IoT, les réseaux intelligents, les réseaux de contrôle et d'acquisition de données en temps réel (SCADA) et les infrastructures des villes intelligentes [5]. Cela a permis d'améliorer la détection, de réduire les temps de réponse et de renforcer la résilience face aux cybermenaces.

Toutefois, l'intégration de ces technologies demeure récente, et c'est pour cette raison que nous cartographions les travaux existants pour évaluer les options de conception et les confronter aux applications réelles. Ici sont présentées les avancées et lacunes de la littérature, tout en mettant en avant les pistes de recherche futures.

2.4. Taxonomie des Jumeaux Numériques

Cette section présente les concepts clés des JN et des composantes nécessaires pour avoir un jumeau. On retrouve plusieurs dimensions techniques et fonctionnelles, selon leur rôle, leur niveau d'intégration, et leur capacité d'analyse. Nous proposons ici une taxonomie adaptée au contexte de la cybersécurité et du ML.

- **1. Déploiement et localisation (D - Déploiement)**
  - (D.1) *Sur site - On-premise* : DT localisé dans l'infrastructure critique.
  - (D.2) *En périphérie - Edge* : Proche des systèmes OT/IoT pour une faible latence.
  - (D.3) *Dans le nuage - Cloud* : Centralisé pour une puissance de calcul.
  - (D.4) *Hybride* : Combinaison Edge-Cloud.
  - (D.5) *Architecture fédérée* : Coordination entre plusieurs DT distribués.
- **2. Niveau de synchronisation du Jumeau Numérique (S - Synchronisation)**
  - (S.1) *Manuelle* : Mise à jour ponctuelle du jumeau par intervention humaine.
  - (S.2) *Périodique - Par jalon* : Synchronisation à intervalles définis, avec état de mise à jour du système.
  - (S.3) *Continue* : Mise à jour en temps réel avec une latence inférieure à la seconde.

  - (S.4) *Boucle fermée* : Couplage dynamique avec retour d'action correctif.
- **3. Rôle dans la sécurité (R - Rôle)**
  - (R.1) *Surveillance passive (monitoring)* : Observation et collecte d'état.
  - (R.2) *Détection d'intrusion (IDS)* : Identification des attaques connues/inconnues.
  - (R.3) *Détection proactive (prédiction)* : Prévision d'attaques avant exploitation.
  - (R.4) *Support décisionnel* : Aide à l'opérateur pour la réponse.
  - (R.5) *Analyse post-incident (forensic)* : Reconstitution après attaque.
- **4. Stratégie d'évolution (E - Evolution)**
  - (A.1) *Statique (modèle figé)* : Entraîné une fois, sans mise à jour.
  - (A.2) *Incrémental* : Mise à jour continue à mesure que de nouvelles données arrivent.
  - (A.3) *Fédéré* : Entraînement réparti entre plusieurs DT ou entités.
- **5. Méthode d'apprentissage (A - Apprentissage)**
  - (A.1) *Supervisé* : Basé sur des données étiquetées.
  - (A.2) *Non-supervisé* : Détection d'anomalies sans étiquettes.
  - (A.3) *Semi-supervisé / Auto-supervisé* : Combinaison partielle d'étiquettes, ou génère ses propres étiquettes.
- **6. Horizon temporel du modèle IA/ML (H - Horizon)**
  - (H.1) *Réactif* : Détection immédiate d'attaques basée sur les flux du système de référence.
  - (H.2) *Estimation court terme* : Estimation d'événements imminents tels que des dérives ou des anomalies.
  - (H.3) *Évaluation à long terme* : Analyse de risque, planification de résilience.

Cette taxonomie permet de positionner les différents travaux existants dans un cadre structuré, facilitant l'analyse comparative et contribuant à identifier les axes de recherche encore peu explorés.

2.5. Architecture de référence

La Figure 1 présente une architecture de référence pour l'intégration des jumeaux numériques et du ML en cybersécurité. Elle comprend quatre couches principales. Chaque couche a un rôle précis pour collecter, modéliser, analyser et exploiter les données afin de détecter et d'anticiper les cyberattaques.

1. **Système cyber-physique** Cette couche représente le système de référence (OT/IT). Elle inclut les capteurs, les actionneurs et les équipements critiques. Elle fournit en continu des données opérationnelles.
2. **Modèle du jumeau numérique** Cette couche assure une représentation dynamique du système physique. Elle maintient une synchronisation en temps réel. Elle permet de simuler des scénarios et de prédire l'état futur du système.
3. **Analyse avec ML et apprentissage** Cette couche applique des algorithmes de ML. Elle analyse les données issues du jumeau numérique. Ses fonctions sont la détection d'anomalies, la corrélation d'événements et la prédiction des risques. Elle utilise des approches supervisées, non supervisées ou hybrides.
4. **Décision et visualisation** Cette couche propose des tableaux de bord et des alertes. Elle fournit des indicateurs de risque et des recommandations. Elle soutient la prise de décision et la planification des actions correctives. En fonction de l'usage voulu, cette couche peut prendre de manière automatique ou avec une validation humaine des décisions, qui seront appliquées via une boucle de rétroaction en direction du système physique.

Cette architecture est modulaire et flexible. Elle peut être déployée sur site, en *Edge* ou dans le *Cloud* selon les besoins en latence, sécurité et calcul. Elle constitue une base générique pour des solutions spécialisées, notamment pour l'estimation des attaques, qui sera discutée plus en détail dans la Section 4.

La Figure 2 présente la structure classique des systèmes industriels ; l'architecture de référence s'appuie sur cette hiérarchie et y ajoute des modules dédiés au JN et au ML. La couche de processus physique correspond au niveau 0 du système ICS. Les niveaux 1 et 2 viennent capturer et transmettre les données nécessaires à la synchronisation du JN et à l'analyse IA. Les couches de ML et décisionnelles de l'architecture de référence viennent compléter ce modèle en introduisant des capacités d'estimation des attaques et d'évaluation des conséquences des attaques, absentes des architectures ICS traditionnelles. La partie décisionnelle est un module supplémentaire au sein du système d'opérations, au sein du niveau 3 dans la partie IT, pour aider les décisionnaires.

Ainsi, la combinaison des deux schémas illustre la similarité entre les infrastructures industrielles existantes et les cadres d'intégration des JN et de l'IA discutés dans la suite de l'article.

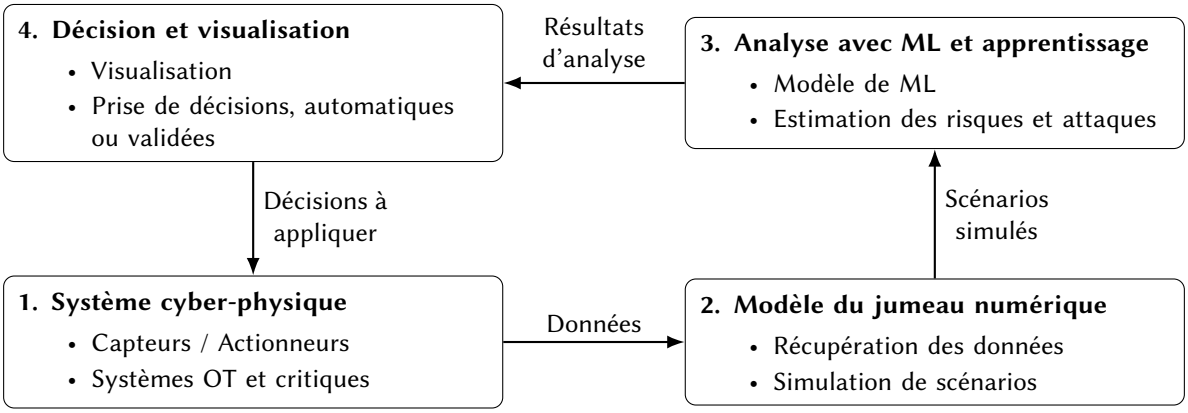

Figure 1: Architecture de référence de l'utilisation du ML et des JN

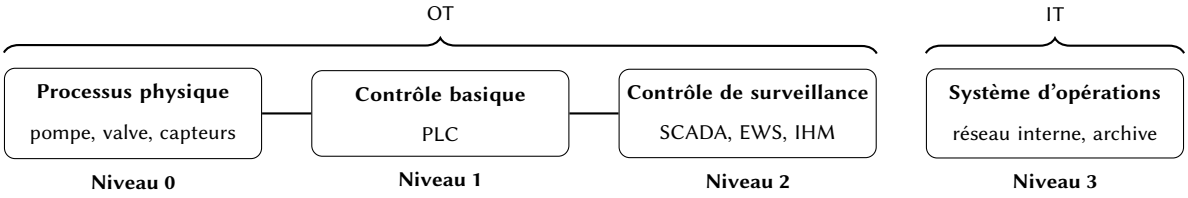

Figure 2: Architecture générale d'un ICS

3. Travaux existants

Pour dresser un panorama des travaux sur l'intégration du ML dans les JN pour la prédiction d'attaques, nous avons analysé les techniques de conception des JN puis les principales applications rapportées dans la littérature. La méthodologie de sélection des articles est détaillée en Section 1. Les résultats sont présentés sous forme de tableaux synthétiques, permettant de comparer les approches et d'identifier les tendances majeures.

3.1. Techniques et implémentations

Cette section présente les aspects techniques et les implémentations liés à la conception des JN dans le cadre de la prédiction et de la détection de cyberattaques. Elle sert à la construction des deux tableaux de synthèse proposés en section 3.2, en détaillant les critères utilisés pour l'analyse des travaux existants.

Les tableaux 1 et 2 offrent une analyse des similitudes entre les implémentations, afin de voir ce qui est couvert. Les deux tables étudient les mêmes articles sur des critères différents ; la première contient des informations sur ce qui est étudié, tandis que la seconde met en avant des indicateurs binaires (oui/non) pour montrer la couverture des travaux. L'analyse de ces dimensions permet de catégoriser les approches existantes et de mettre en évidence les tendances dominantes ainsi que les lacunes actuelles dans l'utilisation des JN pour la cybersécurité.

Les dimensions techniques correspondant aux colonnes des tableaux sont les suivantes. Dans la Table 1, est présenté :

- **Domaine(s) étudié(s)** : Secteur ou environnement cible (IoT, ICS, Smart Grid, etc.).
- **Type de papier** : Revue de littérature, méthode, *framework* ou expérimentation.
- **Rôle attribué au JN** : Simulation, monitoring, analyse, optimisation, détection ou prédiction. Le JN peut avoir plusieurs rôles.
- **Intégration de l'IA-ML** : Nature de l'algorithme ou du modèle employé.
- **Type d'attaque considéré** : Anomalies, intrusions, attaques ciblées ou scénarios simulés.

Tandis que la Table 2 met l'accent sur :

- **Couverture multi-domaines** : Capacité de l'approche à s'appliquer à plusieurs secteurs.
- **Validation par attaques réelles** : Utilisation de scénarios d'attaque concrets pour l'évaluation.
- **Présence d'une prédiction proactive** : Aptitude à anticiper une attaque avant son exploitation.
- **Fonctionnement en circuit fermé** : Existence d'une boucle de rétroaction entre le JN et le système de référence.
- **Validation expérimentale** : Démonstration par mise en œuvre pratique et tests, notamment avec des *testbeds*.

Les lignes des tableaux correspondent aux travaux intégrant le ML dans des JN. Elles reflètent les choix effectués par les auteurs et permettent leur comparaison.

3.2. Comparaison des travaux

Les concepts définis en Section 2.4 servent ici de référence à une analyse comparative des contributions sélectionnées. En pratique, ces dimensions sont appliquées dans les tableaux de synthèse présents au sein de cette section et en section 3.3, afin de positionner chaque étude selon ses caractéristiques techniques et fonctionnelles. Cela permet d'identifier les convergences, les biais de couverture, ainsi que les axes encore peu explorés dans la littérature actuelle. Cette lecture structurée éclaire ainsi les tendances dominantes tout en préparant la discussion sur les verrous et les approches émergentes, développée en Section 4.

Au sein de la Table 1, nous retrouvons deux revues de la littérature, que nous pouvons voir dans la colonne indiquant le type de papier. Les autres papiers sont des implémentations, présentant des *framework* intégrant des modèles de ML. On observe dans la colonne "Intégration de l'IA-ML" que les modèles hybrides de CNN et RNN sont répandus dans les implémentations tandis que XGBoost n'est présent que dans une implémentation de 2024.

Dans la Table 2 on retrouve les mêmes articles étudiés à travers des critères différents afin de mettre en évidence des lacunes de la littérature. Le premier point marquant est le manque d'approches multi-domaines au sein des travaux. Les deux revues de la littérature parlent en effet de plusieurs champs d'application, mais dès que l'on entre dans une partie d'implémentation, le champ est figé à un seul domaine. Ce qui est cohérent avec la taxonomie proposée, mais reste limitant pour appréhender ces travaux dans d'autres domaines.

Table 1
Classification des méthodes de ML et attaques étudiées

Ref	Domaine(s)	Type de papier	Rôle du JN	Intégration de l'IA-ML	Type d'attaque
[6]	Général	Revue de la littérature	Optimisation, prédiction	Discussion IA/ML sans technique spécifique implémentée	Prédiction d'incidents (revue de l'état de l'art / outils)
[7]	Général	Revue de la littérature	Simulation, monitoring, analyse	Prédiction holistique, augmentation de la maturité	- (Revue)
[8]	IoT	Méthode, Framework, Expérimentation	Simulation	Hybride CNN+RNN pour détection et prédiction d'anomalies/menaces	Détection et prédiction (simulation, anticipation, réaction)
[9]	SCADA, Smart Grid	Méthode, Framework	Visualisation, prédiction	ML hybride (algorithme non précisé) : visualisation et prédiction des effets	Visualisation et prédiction des conséquences d'attaque
[10]	Smart Infrastructure Networks	Framework, Expérimentation	Surveillance, prédiction	Autoencodeur (apprentissage de caractéristiques) + RNN (séquences)	Détection, analyse prédictive, détection d'anomalies
[11]	Smart City (city subsystems)	Framework, Expérimentation	Simulation	CNN-LSTM entraîné sur CICIDS2017	Détection, prédiction, évaluation des attaques
[12]	ICS (Industrial Filling Plant)	Framework, Expérimentation	Simulation, détection	Autoencodeurs, RNN	Détection et classification d'intrusions
[13]	IoT	Framework, Expérimentation	Surveillance	Détection d'anomalies, analyse prédictive	Détection et sécurité proactive (dite prédictive)
[14]	ICS (distribution d'eau)	Framework, Expérimentation	Simulation	Hybride CNN & RNN	Détection d'attaques
[15]	CPS génériques	Expérimentation	Simulation	PCA, autoencodeur, Random Forest, XGBoost	Détection d'intrusions

Cependant, toutes les études présentant des *frameworks* n'ont pas fait uniquement des travaux théoriques mais ont bien implémenté ce qu'ils présentent et l'ont validé par l'expérimentation. Au vu des résultats obtenus, un potentiel dans l'utilisation des JN avec le ML pour la prédiction d'attaques se remarque.

3.3. Positionnement dans la taxonomie

La Table 3 présente le positionnement des contributions recensées au sein de la taxonomie définie en Section 2.4, en les répartissant selon les dimensions *Déploiement*, *Synchronisation*, *Rôle*, *Évolution*, *Apprentissage* et *Horizon*. Les travaux [7] et [6], qui sont des revues de littérature, ne figurent pas dans la Table 3 car elle se limite aux implémentations.

La comparaison des travaux recensés au sein de la Table 3 met en évidence une concentration des efforts sur des JN reposant sur une *Synchronisation Continue* (S.3) entre le système physique et son modèle virtuel. Cela reflète la volonté des auteurs de maximiser la réactivité et la précision de la détection, mais une telle synchronisation implique en contrepartie des exigences élevées en matière de bande passante, de fiabilité des communications et de coûts de traitement.

Ces JN sont le plus souvent couplés à un usage *Réactif* du modèle de ML (H.1), focalisé sur la détection

Table 2
Comparaison des approches intégrant les JN et le ML pour la détection et la prédiction d'attaques

Ref	Multi-domaines	Évalué avec attaques réelles	Prédiction proactive	JN en circuit fermé	Validé avec expérimentation
[6]	Oui	Non	Non	Non	Non
[7]	Oui	Non	Non	Non	Non
[8]	Non	Oui	Oui	Oui	Oui
[9]	Non	Non	Oui	Non	N/A
[10]	Non	Non	Non	Oui	Oui
[11]	Non	Oui	Oui	Non	Oui
[12]	Non	Oui	Non	Oui	Oui
[13]	Non	Non	Oui	Oui	Oui
[14]	Non	Oui	Non	N/A	Oui
[15]	Non	Oui	Non	N/A	Oui

immédiate d'attaques ou d'anomalies au moment où elles surviennent. Cette orientation s'explique par la facilité d'évaluation et la disponibilité de données annotées. Elle tend à limiter les capacités prédictives et la prise en compte d'évolutions à moyen ou long terme, car les modèles entraînés sur des données annotées se focalisent sur des comportements passés et peinent à anticiper des situations nouvelles [7].

De manière cohérente, la majorité des approches adoptent un *Apprentissage Supervisé* (A.1), tirant parti de jeux de données étiquetés pour entraîner les modèles. Si cette stratégie permet d'obtenir de bonnes performances en détection de comportements connus, elle reste dépendante de la qualité et de la représentativité des données disponibles, ce qui limite son efficacité face à des scénarios inédits [6].

Enfin, ces contributions s'appuient très largement sur une *Évolution Statique* (E.1), où les modèles sont entraînés une fois puis appliqués sans mise à jour significative. Cette approche simplifie la mise en œuvre et réduit la complexité de calcul, mais elle limite la capacité du système à s'adapter à l'évolution des menaces et aux changements de contexte opérationnel.

Les implémentations avec une logique *Incrémentale* (E.2) ou visant un *Horizon Évaluatif long terme* (H.3) sont absentes dans les études recensées bien que souvent indiquées comme piste de recherche future [9]. De plus, les déploiements sont majoritairement limités à des environnements *Sur site* (D.1) ou *Cloud* (D.3), avec peu d'explorations d'approches *Hybrides* (D.4). Le lien avec l'apprentissage fédéré (D.5) n'apparaît pas dans les travaux étudiés.

Cette cartographie suggère un écart entre le potentiel théorique du paradigme JN couplé au ML et ses mises en œuvre actuelles, soulignant la nécessité d'explorer des architectures plus dynamiques, résilientes et généralisables.

4. Verrous actuels et approches émergentes

4.1. Vers un Jumeau Numérique prédictif

Le concept de JN appliqué à la cybersécurité est encore majoritairement cantonné à une logique descriptive, c'est-à-dire des approches réactives centrées sur la duplication de flux ou la surveillance d'états (S.3, H.1). Les architectures actuelles se concentrent souvent sur la duplication de flux ou la surveillance de métriques, sans capturer la dynamique complète, soit l'évolution du système et de ses états internes. Par exemple, un JN doit pouvoir représenter la propagation d'une variation de débit dans un réseau d'eau, ainsi que les relations structurelles, comme les dépendances entre capteurs, actionneurs et contrôleurs dans un ICS. Face à des menaces de plus en plus furtives, il devient important de dépasser cette approche réductrice, difficile à détecter par des mesures instantanées. Les synthèses confirment que les architectures réellement anticipatives (S.4, H.3) restent peu démontrées et souvent monosectorielles [7] [6]. Plusieurs travaux annoncent la "prédiction", mais implémentent surtout une

Table 3
Positionnement des travaux recensés dans la taxonomie

Ref	Déploiement	Synchronisation	Rôle sécurité principal	Stratégie d'évolution	Méthode d'apprentissage	Horizon IA/ML
[8]	Sur site	Continue	Détection d'intrusion	Statique	Supervisé	Réactif
[9]	En périphérie	Continue	Détection proactive	Statique	Supervisé	Estimation court terme
[10]	Sur site	Continue	Détection d'intrusion	Statique	Non-supervisé	Réactif
[11]	Hybride	Périodique	Support décision	Statique	Supervisé	Estimation court terme
[12]	Nuage	Continue	Détection d'intrusion	Statique	Semi-supervisé	Réactif
[13]	Nuage	Continue	Détection proactive	Statique	Supervisé	Estimation court terme
[14]	En périphérie / Hybride	Continue	Détection proactive	Statique	Non-supervisé	Estimation court terme
[15]	Nuage	Périodique	Détection d'intrusion	Statique	Supervisé	Réactif

analyse d'anomalies en temps réel. Notamment avec le modèle de JN pour la cybersécurité utilisant un modèle de ML avec CNN et RNN [9] ou avec des autoencodeurs d'infrastructures intelligentes [12]. On retrouve aussi des frameworks IoT dits prédictifs mais centrés sur l'anomalie [14]. À l'inverse, les approches explicitant un modèle du système de référence au sein du JN, plus rares, ouvrent la voie à un JN permettant d'estimer les attaques car il ne se contente pas de détecter des anomalies présentes, mais projette leurs conséquences possibles. Par exemple, un jumeau SCADA fondé sur un modèle basé sur un graphe de connaissances, représentant les relations entre composants et attaques [11].

4.1.1. JN comme miroirs de données

Dans les travaux référencés, le JN se limite à recopier des flux réseau ou des métriques sans représenter les états internes du système protégé. On obtient un miroir de données utile à la surveillance, mais peu apte à raisonner sur des trajectoires futures ou des scénarios hypothétiques. Par exemple, le cyber-twin IoT de Jyothi et al. [9] effectue la détection d'anomalies sur des trames en temps réel, mais n'implémente aucun modèle structurel de l'objet connecté ni de son environnement opérationnel. De même, la plate-forme ICS de Varghese et al. [8] se concentre sur l'injection de scénarios d'attaque dans une maquette logicielle, sans couplage étroit avec les lois physiques du procédé. Ces JN-miroirs optimisent la détection et la réactivité, en améliorant la rapidité de l'identification d'anomalies présentes. Cependant, ils n'offrent ni estimation probabiliste à horizon (H.2, H.3), ni capacité d'expliquer les alertes par la modélisation du comportement du système. Pour dépasser ce rôle descriptif, il est nécessaire d'intégrer des modèles physico-logiques capables d'estimer des probabilités d'évolution et de relier chaque alerte à une trajectoire attendue.

4.1.2. Manque de modèles physico-logiques de référence

Les revues de l'état de l'art soulignent que la plupart des DT de cybersécurité restent conceptuels et n'intègrent ni équations de comportement ni contraintes de contrôle pour refléter fidèlement le système réel [7]. On observe cependant une volonté de représenter ce comportement visible dans certaines études. Avec l'approche hybride-automate de Pisani et al. [10], qui modélise la couche physique et la couche de commande d'un réseau d'eau industriel. Toutefois, cette démonstration reste isolée et monosectorielle, mais demeure une piste intéressante montrant l'importance de la modélisation explicite des systèmes au sein des JN. L'étude de [11] s'appuie sur une modélisation sous forme de graphe de connaissance

afin de représenter trois types d'attaques et d'évaluer leurs effets sur la sûreté et la disponibilité. Ces exemples soulignent un besoin de généraliser l'intégration de modèles physico-logiques.

4.1.3. Nécessité d'une modélisation explicite

Pour construire un JN permettant de faire de l'estimation proactive, il est nécessaire d'inclure une modélisation explicite du système de référence et de ne pas se limiter à une agrégation de données envoyées à un modèle de ML. La modélisation doit inclure les composantes du système permettant de représenter son état interne et étant pertinentes pour simuler l'évolution du système. Cette modélisation peut prendre la forme de modèles physiques avec des équations mathématiques, des modèles de contrôle avec des séquences ou des consignes pour des automates, des règles de supervision, des graphes d'interdépendance, etc. Les données d'entrée viennent alimenter les modèles qui alimentent et actualisent la modélisation. C'est uniquement par-dessus que vient se superposer un modèle de ML qui peut estimer certains comportements, que ce soit pour l'analyse en temps réel ou pour la simulation de scénarios. Cette modélisation permet de filtrer plus tôt les faux positifs par cohérence physique, d'identifier les variables critiques qu'un attaquant devrait manipuler, et d'expliquer les alertes au travers de l'écart entre comportement simulé et observé. Les écarts de comportement peuvent être dus à des déviances ou à l'oubli de certains paramètres dans la modélisation. La modélisation physique est d'autant plus importante dans l'ICS d'eau présent dans [10] où apparaît un décalage persistant du débit malgré une injection de fausses données ou une action effectuée. Comme le montrent les auteurs, cet écart est dû à la masse de l'eau déplacée, à l'impact sur la commande des vannes et à la variation de vitesse engendrée.

4.1.4. Anticiper les menaces par simulation prospective

La boucle simulation–apprentissage d'un jumeau numérique prédictif s'articule autour de la génération de scénarios, de la propagation, de l'apprentissage et de la décision. Le jumeau produit d'abord des scénarios hypothétiques couvrant différentes tactiques, techniques et procédures (TTP), comme l'injection de paquets malveillants ou l'altération de capteurs. Ces scénarios sont ensuite propagés à travers les modèles physico-logiques et de contrôle afin de générer des trajectoires simulées et d'estimer les distributions de temps-à-risque ou d'impact. Le pipeline d'apprentissage automatique exploite alors ces séquences étiquetées, issues d'un mélange de données réelles validées et de données synthétiques, pour assurer un apprentissage incrémental et un durcissement adversarial. Le durcissement renforce le modèle en l'exposant à des attaques artificielles afin d'améliorer sa robustesse. Enfin, le système produirait des sorties orientées vers la prévision, telles que des probabilités ou des indices d'impact à horizon donné, et permettrait d'évaluer dans le jumeau les contre-mesures candidates avant leur déploiement en conditions réelles. Ce processus méthodologique vise ainsi à transformer la surveillance réactive en une prévention proactive, en exploitant les écarts entre comportements simulés et observés comme signal central pour l'entraînement et la prise de décision.

À l'état de l'art, plusieurs travaux explorent certaines briques de cette chaîne sans en proposer une mise en œuvre complète. On retrouve par exemple des approches combinant injection de scénarios et détection en temps réel dans des jumeaux numériques dédiés aux systèmes industriels [8]. D'autres initiatives s'appuient sur des jumeaux SCADA couplés à des graphes de connaissances afin de visualiser et d'estimer à l'avance les effets d'attaques, mais sans démontrer de mécanismes de mise à jour incrémentaux et réguliers du modèle prédictif [11]. Les revues récentes confirment par ailleurs que la prévision à horizon, la mise à jour incrémentale en ligne et la validation multi-domaine restent largement au stade conceptuel et ne sont que rarement documentées dans des systèmes opérationnels combinant jumeaux numériques et intelligence artificielle [9, 8]. Ces manques orientent les perspectives et constituent le socle des contributions futures.

4.2. Pipeline de ML pour la cybersécurité

L'intégration de ML dans les JN pour la cybersécurité reste, à ce jour, marquée par des pratiques statiques et cloisonnées à un domaine. Alors que le potentiel de l'IA pour anticiper des comportements anormaux

ou détecter des signaux faibles est reconnu [7], les implémentations concrètes peinent à dépasser des approches classiques basées sur des modèles statiques, rarement mis à jour. Cette section met en lumière les principales limites observées dans la littérature avant de présenter une approche palliant ces lacunes et adaptée aux environnements dynamiques et aux contraintes des systèmes industriels.

4.2.1. Modèles figés

Les systèmes de sécurité basés sur des JN entraînent leurs classificateurs hors du système, puis figent les poids une fois déployés. L'étude d'un ICS sélectionne un empilement après évaluation hors-ligne de huit algorithmes, l'adaptation au fil du temps n'est pas prévue [8]. Des tendances analogues se retrouvent dans des infrastructures IoT et smart city [9] [13]. Sans apprentissage incrémental, les modèles ne suivent pas l'évolution des TTP attaquantes et la protection se dégrade au fil du temps.

4.2.2. Absence de pipeline d'Inférence / Mise à jour dédié

Les cadres existants exécutent la détection et parfois le ré-entraînement dans le même fil d'exécution, voire n'en parlent pas. Le « cyber-twin » pour l'IoT, par exemple, fait transiter le trafic en temps réel dans un détecteur CNN-RNN, mais ne décrit jamais de copie fantôme ni de procédure de bascule à chaud pour les mises à jour du modèle [9]. L'utilisation de données dont la provenance est non vérifiée augmente aussi l'exposition à la contamination par des données bruitées ou malveillantes lors d'apprentissages en ligne non contrôlés [16]. L'absence de registres de modèles versionnés et de caractéristiques de ML fixes, en lecture seule, compromet la traçabilité des versions de modèles et freine la transition d'une logique purement réactive vers des capacités prédictives validées à l'horizon (H.3). À l'échelle du domaine, ce manque d'ingénierie des mises à jour enferme les approches dans un contexte unique et empêche leur adaptation fiable à d'autres secteurs, d'où le déficit de généralisation relevé par les synthèses [7] [6].

4.2.3. Détection réactive plutôt que prédiction prospective

Les synthèses de l'état de l'art indiquent que la plupart des travaux DT–IA évaluent encore le succès via des métriques de classification instantanée sur des événements présents, sans formuler ni valider une prévision à horizon explicite. La véritable prévision à l'horizon futur demeure conceptuelle et sans validation pratique [7]. Les défenseurs découvrent ainsi l'attaque lorsqu'elle est déjà en cours, au lieu de recevoir une trajectoire de confiance pour les prochaines minutes ou heures. Pour passer d'une logique réactive à une prédiction utilisable opérationnellement, les sorties du modèle devraient fournir une trajectoire de confiance sur plusieurs horizons pondérée par un indice d'impact issu du jumeau physico-logique afin d'évaluer la gravité des menaces anticipées.

4.2.4. Apprentissage et inférence séparés : un cadre de double usage

Parmi les approches envisagées pour dépasser les limites des JN actuels, un mode de double utilisation intégrant une séparation claire entre apprentissage et inférence est une approche prometteuse. Concrètement, un même modèle de ML opère en lecture seule, dédié à l'inférence en temps réel sur des données non vérifiées. En parallèle, il est possible de mettre à jour le modèle en continu à partir de données jugées fiables, issues d'observations réelles validées, c'est-à-dire dont on connaît la provenance et l'authenticité. Ces données peuvent également provenir de jeux enrichis avec des scénarios d'attaque simulés.

Ce découplage permet d'éviter toute contamination du modèle de ML par des données bruitées ou malveillantes, tout en assurant une adaptation incrémentale à l'évolution des menaces. Une fois que la nouvelle version du modèle a été testée et validée sur un jeu de données dédié, elle peut être déployée à chaud sans perturber le fonctionnement opérationnel. Ce mécanisme contribue à faire évoluer la logique réactive de détection vers une capacité prédictive plus proactive. Le double training améliore la généralisation inter-domaine, en effet il introduit un apprentissage incrémental à partir de données hétérogènes tout en préservant la stabilité du modèle opérationnel.

4.3. Conclusion

Ces éléments mettent en évidence deux constats majeurs. Premièrement, que les approches actuelles restent trop réactives et figées et deuxièmement, que les pipelines ML manquent de mécanismes de mise à jour incrémentale. En réponse, l'article met en avant la nécessité d'une modélisation explicite du système de référence au sein du JN et d'un mode de double usage du modèle de ML permettant de combiner simulation prospective et apprentissage continu. Cette orientation ouvre la voie vers des JN plus précis et adaptatifs, mieux alignés avec les exigences des systèmes critiques.

5. Challenges

Cette section synthétise les principaux défis liés à l'intégration des JN et du ML pour la cybersécurité. Bien que l'intégration des JN et du ML offre des bénéfices significatifs, elle engendre également des problématiques, tant de manière individuelle que lorsqu'ils sont combinés. Ces difficultés sont amplifiées par la complexité des systèmes industriels et des environnements dans lesquels ils évoluent.

Le Table 4 propose une synthèse des principaux défis identifiés, de leurs impacts potentiels et des solutions envisagées dans notre approche. Les sous-sections qui suivent détaillent chacun de ces points.

Table 4
Synthèse des challenges liés à l'intégration JN et ML

Challenge	**Contrainte principale**	**Impact potentiel**	**Solutions envisagées**
Qualité des données	ML	Biais ou erreurs dans les prédictions, mauvais entraînement du modèle	Sélection conditionnelle des données fiables, génération de données simulées réalistes
Modélisation réaliste du JN	ICS/CPS	Décalage entre le système réel et le modèle virtuel, perte de précision des estimations	Utilisation de données HITL, ajustement progressif par boucles de rétroaction
Robustesse des modèles de ML	ML et Cybersécurité	Vulnérabilité aux attaques *adversariales*, déviation du modèle	Apprentissage incrémental, détection d'anomalies génériques, intégration future d'IA explicable
Contraintes temps réel et latence	ICS/CPS et Cybersécurité	Décisions trop lentes, inefficacité en contexte critique	Déploiement hybride Edge-Cloud, traitement local pour les décisions urgentes
Interopérabilité industrielle	ICS/CPS	Difficulté d'intégration dans les infrastructures OT existantes	Usage de protocoles standards comme Modbus TCP/IP, compatibilité avec PLC ouverts type OpenPLC
Complexité du pipeline JN+IA	ML et CPS/ICS	Maintenance difficile, vérification du bon fonctionnement global	Architecture modulaire, séparation des couches

5.1. Les données

Les problématiques liées aux données concernent à la fois les JN et le ML. Un axe commun concerne la collecte, le stockage et l'exploitation des données. Pour le JN, ces défis se manifestent dans la nécessité d'assurer une synchronisation fiable et un fonctionnement en temps réel. Pour le ML, il se traduit par l'exigence d'un jeu de données représentatif, complet, exempt de bruit et correctement annoté, indispensable pour entraîner un modèle capable d'estimer ou de détecter avec précision.

Un enjeu central est de garantir des données représentatives, fiables et diversifiées. C'est sur ce point que plusieurs travaux soulignent une limite importante : la constitution de jeux de données réalistes reste très incomplète. Par exemple, aucun outil ne permet de générer un jeu de données couvrant l'ensemble du périmètre d'une attaque complexe [17].

5.2. Défis liés à la validation

La validation des approches JN–ML représente un défi, car les résultats obtenus dépendent autant du réalisme du jumeau numérique que de la méthodologie d'évaluation employée. Que ce soit pour l'estimation d'attaque ou la détection, un modèle de ML entraîné sur un JN ne garantit en rien qu'il sera efficace sur le système réel. En fonction des paramètres de modélisation choisis, le système respectera un niveau de réalisme défini. Un risque réside dans le fait que les performances observées ne soient qu'un reflet des approximations ou lacunes du JN. Une approche permettant de vérifier ce qui est fait est la notion d'IA explicable (*Explainable AI*), qui permet d'identifier les paramètres influençant les décisions du modèle et de s'assurer qu'elles reposent sur des signaux légitimes plutôt que sur des biais ou des artefacts du JN. Cette transparence est particulièrement importante en cybersécurité, où une mauvaise interprétation peut conduire à ignorer une attaque réelle ou à générer de faux positifs. Une piste complémentaire consiste à recourir à des jeux de données hybrides, mêlant observations réelles et scénarios simulés par le JN, afin d'évaluer la robustesse du modèle dans des conditions variées. Des validations croisées sur bancs d'essai permettent également de réduire le risque de sur-adaptation. Dans le contexte JN–ML, un modèle trop ajusté aux données simulées peut sembler très performant en test, mais échouer à détecter une attaque réelle dès que les conditions diffèrent légèrement de celles du jumeau ou du jeu d'entraînement

Une revue systématique sur les tests basés sur JN pour les CPS souligne que le jumeau sert souvent pour les tests et que la validité des évaluations dépend des spécifications et des données qui ont servi à construire le jumeau lui-même, avec une capacité prédictive encore limitée dans les travaux existants [18]. Cela renforce le risque d'un écart entre JN et le réel et justifie l'usage de jeux de données hybrides (réel et simulé), de validations croisées sur bancs d'essai et d'outils d'explicabilité pour auditer ce que le modèle a effectivement appris.

5.3. Robustesse et sécurité des modèles ML

Les modèles de ML sont vulnérables aux attaques dites *adversariales*, où des données malveillantes peuvent perturber leur comportement ou tromper leurs prédictions [19] [20]. Dans un contexte CPS/ICS, ce risque est accentué par des flux de données dont l'authenticité ou l'intégrité n'est pas toujours garantie. Il devient alors essentiel de contrôler quelles données participent à l'apprentissage, afin d'éviter une contamination du modèle.

Une autre difficulté réside dans la robustesse face aux scénarios non connus (*zero-day attacks*). Un modèle de ML entraîné sur un jeu limité ne généralise pas toujours correctement, ce qui pose des risques en production. Une piste envisagée dans la littérature est l'adoption d'un apprentissage incrémental, permettant au modèle de s'adapter aux évolutions des menaces tout au long de son cycle de vie [9] [21] [20]. Contrairement aux modèles figés, cette approche réduit le risque d'obsolescence face aux attaques inédites. Elle reste toutefois sensible aux données bruitées ou malveillantes, ce qui impose des mécanismes de contrôle adaptés.

Une piste complémentaire réside dans l'intégration de techniques d'intelligence artificielle explicable, afin de mieux comprendre les décisions prises par les modèles de ML dans des contextes critiques. Des méthodes comme SHapley Additive exPlanations (SHAP) ou Local Interpretable Model-agnostic Explanations (LIME) permettent d'identifier quelles caractéristiques influencent le plus la décision du modèle. Par exemple, SHAP aide à identifier les variables déterminantes qui ont conduit à la classification d'un comportement. Ces approches renforcent la confiance dans les prédictions, car elles rendent les décisions moins opaques et permettent de vérifier leur cohérence avec des connaissances expertes. Elles facilitent également la validation du modèle par des experts métiers, en leur donnant des éléments interprétables. Par exemple, le modèle peut indiquer l'importance relative d'une variable physique ou d'un indicateur réseau, plutôt que de fournir uniquement une sortie binaire de type "attaque/non-attaque" [21].

5.4. Intégration dans des environnements industriels

L'intégration d'un JN et de modèles ML dans des environnements industriels impose des contraintes supplémentaires :

- **Latence et temps réel :** Le traitement des données et les prédictions doivent respecter des délais stricts, surtout pour les systèmes critiques.
- **Interopérabilité :** Les protocoles industriels (Modbus, OPC-UA, etc.) et les infrastructures OT n'ont pas été initialement conçus pour interagir avec des solutions de ML, telles que l'apprentissage profond.
- **Maintenance et mise à jour :** Les modèles ML doivent être continuellement réentraînés pour rester efficaces, ce qui peut être complexe en environnement opérationnel.

Ces contraintes montrent que l'intégration JN–ML dans l'industrie ne peut se limiter à un simple transfert de technologies académiques, mais doit s'accompagner d'adaptations architecturales et organisationnelles, afin de concilier performances, sûreté et continuité de service.

5.5. Complexité de la combinaison JN-ML

La combinaison d'un JN et du ML augmente la complexité des pipelines de données, de la synchronisation et de l'évaluation des performances. Un écart JN-réel peut entraîner des erreurs de prédiction. Il est donc nécessaire de définir des mécanismes de contrôle, d'audit et de résilience pour assurer la fiabilité du système global.

Dans un environnement industriel, un JN hybride combinant modèles physiques et automates, associé à un empilement de modèles ML, illustre bien la complexité opérationnelle. Le pipeline doit gérer des flux hétérogènes issus du procédé et du réseau, dont la synchronisation incomplète ou retardée peut provoquer des divergences entre système réel et modèle virtuel. Une modification mineure non répercutée dans le jumeau peut générer des écarts persistants, amplifiés par le ML et interprétés à tort comme une attaque. Cela souligne la nécessité de mécanismes de synchronisation fiables pour la combinaison JN–ML.

6. Conclusion

Cet article présente une synthèse des travaux réalisés sur les JN et leur intégration avec le ML dans le domaine de la cybersécurité des systèmes industriels. Nous avons analysé les verrous actuels et présenté des approches émergentes, notamment en direction d'architectures prédictives basées sur des JN (Section 4.1). Cette approche combine la modélisation dynamique du JN avec des techniques de ML hybrides (Section 4.2), tout en mettant l'accent sur la détection proactive et l'évaluation des impacts.

Ce travail met en évidence la nécessité de disposer de jeux de données fiables (Section 5.1), essentiels tant pour l'entraînement des modèles de ML que pour la validation des JN dans des conditions représentatives (Section 5.2). Le mécanisme de sélection conditionnelle de l'apprentissage, fondé sur un filtrage explicite des données fiables, constitue une approche prometteuse pour renforcer la robustesse et l'adaptabilité des modèles (Section 5.3) dans un contexte industriel. Ces deux propriétés sont essentielles pour suivre l'évolution des menaces et du contexte opérationnel, tout en garantissant la pérennité et l'efficacité du système à long terme.

Enfin, des pistes de recherche restent ouvertes (Section 5) concernant la qualité et la diversité des données utilisées, le réalisme des JN et les méthodes de validation à grande échelle. Ces perspectives constituent des étapes clés pour renforcer la fiabilité et favoriser l'adoption de ces approches dans des environnements industriels physiques.

Remerciements

Cette recherche est financée par la chaire de recherche industrielle Cybersécurité des Infrastructures Critiques (CyberCNI) et par l'université du Québec à Chicoutimi (UQAC).

Déclaration sur l'IA générative

En préparant ce travail, les auteurs ont utilisé de l'IA, notamment Grammarly, uniquement pour corriger les fautes de grammaire et d'orthographe. Après utilisation de ces outils, le travail a été revu et les auteurs prennent la pleine responsabilité du contenu de la publication.

Declaration on Generative AI

During the preparation of this work, the authors used Grammarly for grammar and spelling correction. After using this tool, the work was reviewed, and the authors take full responsibility for the publication's content.

References

[1] T. Kreuzer, P. Papapetrou, J. Zdravkovic, Artificial intelligence in digital twins—A systematic literature review, Data & Knowledge Engineering 151 (2024) 102304. URL: https://www.sciencedirect.com/science/article/pii/S0169023X24000284. doi:10.1016/j.datak.2024.102304.

[2] M. Grieves, J. Vickers, Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems, in: F.-J. Kahlen, S. Flumerfelt, A. Alves (Eds.), Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches, Springer International Publishing, Cham, 2017, pp. 85–113. URL: https://doi.org/10.1007/978-3-319-38756-7_4. doi:10.1007/978-3-319-38756-7_4.

[3] Q. Xu, S. Ali, T. Yue, Digital Twin-based Anomaly Detection in Cyber-physical Systems, in: 2021 14th IEEE Conference on Software Testing, Verification and Validation (ICST), 2021, pp. 205–216. URL: https://ieeexplore.ieee.org/document/9438560. doi:10.1109/ICST49551.2021.00031, iSSN: 2159-4848.

[4] A. Murillo, R. Taormina, N. Tippenhauer, S. Galelli, Co-Simulating Physical Processes and Network Data for High-Fidelity Cyber-Security Experiments, in: Sixth Annual Industrial Control System Security (ICSS) Workshop, ICSS 2020, Association for Computing Machinery, New York, NY, USA, 2021, pp. 13–20. URL: https://doi.org/10.1145/3442144.3442147. doi:10.1145/3442144.3442147.

[5] K. Ramesh, S. GVK, M. Rao, J. Bapat, D. Das, Digital Twin based What-if Simulation of Security Attacks in Smart Irrigation Systems, in: 2024 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), 2024, pp. 1–6. URL: https://ieeexplore.ieee.org/document/10677126. doi:10.1109/CONECCT62155.2024.10677126, iSSN: 2766-2101.

[6] A. Pokhrel, V. Katta, R. Colomo-Palacios, Digital Twin for Cybersecurity Incident Prediction: A Multivocal Literature Review, in: Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops, ICSEW'20, Association for Computing Machinery, New York, NY, USA, 2020, pp. 671–678. URL: https://doi.org/10.1145/3387940.3392199. doi:10.1145/3387940.3392199.

[7] J. Luzzi, R. Naha, A. Arulappan, A. Mahanti, SoK: A Holistic View of Cyberattacks Prediction with Digital Twins, in: Second International Conference on Emerging Trends in Information Technology and Engineering (ICETITE), 2024, pp. 1–7. URL: https://ieeexplore.ieee.org/document/10493514. doi:10.1109/ic-ETITE58242.2024.10493514.

[8] S. A. Varghese, A. Dehlaghi Ghadim, A. Balador, Z. Alimadadi, P. Papadimitratos, Digital Twin-based Intrusion Detection for Industrial Control Systems, in: IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), 2022, pp. 611–617. URL: https://ieeexplore.ieee.org/document/9767492. doi:10.1109/PerComWorkshops53856.2022.9767492.

[9] R. Jyothi, R. Jagadeesha, Next-Gen Threat Detection: Leveraging AI and Cyber Twin Technologies for IoT Security, in: First International Conference on Software, Systems and Informa-

tion Technology (SSITCON), 2024, pp. 1–6. URL: https://ieeexplore.ieee.org/document/10796384. doi:10.1109/SSITCON62437.2024.10796384.

[10] J. Pisani, G. Cavone, F. Pascucci, L. Giarré, Using Digital Twin to Detect Cyber-Attacks in Industrial Control Systems, in: IEEE EUROCON - 20th International Conference on Smart Technologies, 2023, pp. 467–471. URL: https://ieeexplore.ieee.org/document/10198927. doi:10.1109/EUROCON56442.2023.10198927.

[11] N. Al-Qirim, M. Majdalawieh, A. Bani-hani, H. Al Hamadi, Cyber threat intelligence for smart grids using knowledge graphs, digital twins, and hybrid machine learning in SCADA networks, International Journal of Engineering Business Management 17 (2025) 18479790251328183. URL: https://doi.org/10.1177/18479790251328183. doi:10.1177/18479790251328183, publisher: SAGE Publications Ltd STM.

[12] M. Sasikala, Y. M. Mahaboob John, B. Jothi, Nandhini S, Senthil Kumar S, Integrating Digital Twins with AI for Real-Time Intrusion Detection in Smart Infrastructure Networks, in: International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS), 2024, pp. 1–6. URL: https://ieeexplore.ieee.org/document/10721892. doi:10.1109/IACIS61494.2024.10721892.

[13] A. Mayan, S. Krishanveni, B. Jothi, AI Enabled Digital Twin Models to Enhance Security in Smart Cities, in: International Conference on Intelligent Computing and Sustainable Innovations in Technology (IC-SIT), 2024, pp. 1–6. URL: https://ieeexplore.ieee.org/document/10862955. doi:10.1109/IC-SIT63503.2024.10862955.

[14] K. S. Supriya, Jeno Lovesum S P, R. Arora, R. Bhatia, S. Yadwad, L. Natrayan, Securing loT Systems with AI-Infused Software and Virtual Replica Models, in: International Conference on Integrated Intelligence and Communication Systems (ICIICS), 2024, pp. 1–6. URL: https://ieeexplore.ieee.org/document/10860178. doi:10.1109/ICIICS63763.2024.10860178.

[15] H. C. Ukwuoma, G. Dusserre, G. Coatrieux, J. Vincent, N. B. Ahmed, Optimising Intrusion Detection in Cyber-Physical Systems, in: 8th Cyber Security in Networking Conference (CSNet), 2024, pp. 7–14. URL: https://ieeexplore.ieee.org/document/10851766. doi:10.1109/CSNet64211.2024.10851766, iSSN: 2768-0029.

[16] Z. Wang, J. Ma, X. Wang, J. Hu, Z. Qin, K. Ren, Threats to Training: A Survey of Poisoning Attacks and Defenses on Machine Learning Systems, ACM Comput. Surv. 55 (2022) 134:1–134:36. URL: https://dl.acm.org/doi/10.1145/3538707. doi:10.1145/3538707.

[17] C. Lo, T. Y. Win, Z. Rezaeifar, Z. Khan, P. Legg, Digital Twins in Industry 4.0 Cyber Security, in: IEEE Smart World Congress (SWC), 2023, pp. 1–4. URL: https://ieeexplore.ieee.org/document/10449147. doi:10.1109/SWC57546.2023.10449147.

[18] R. J. Somers, J. A. Douthwaite, D. J. Wagg, N. Walkinshaw, R. M. Hierons, Digital-twin-based testing for cyber–physical systems: A systematic literature review, Information and Software Technology 156 (2023) 107145. doi:10.1016/j.infsof.2022.107145.

[19] A. Vassilev, A. Oprea, A. Fordyce, H. Anderson, X. Davies, M. Hamin, Adversarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations, Technical Report NIST Artificial Intelligence (AI) 100-2 E2025, National Institute of Standards and Technology, 2025. URL: https://csrc.nist.gov/pubs/ai/100/2/e2025/final. doi:10.6028/NIST.AI.100-2e2025.

[20] B. Steenwinckel, D. De Paepe, S. Vanden Hautte, P. Heyvaert, M. Bentefrit, P. Moens, A. Dimou, B. Van Den Bossche, F. De Turck, S. Van Hoecke, F. Ongenae, FLAGS: A methodology for adaptive anomaly detection and root cause analysis on sensor data streams by fusing expert knowledge with machine learning, Future Generation Computer Systems 116 (2021) 30–48. URL: https://www.sciencedirect.com/science/article/pii/S0167739X20329927. doi:10.1016/j.future.2020.10.015.

[21] I. H. Sarker, H. Janicke, A. Mohsin, A. Gill, L. Maglaras, Explainable AI for cybersecurity automation, intelligence and trustworthiness in digital twin: Methods, taxonomy, challenges and prospects, ICT Express 10 (2024) 935–958. URL: https://www.sciencedirect.com/science/article/pii/S2405959524000572. doi:10.1016/j.icte.2024.05.007.